

---

# Extracting Certainty from Uncertainty: Regret Bounded by Variation in Costs

---

**Elad Hazan**

IBM Almaden

650 Harry Rd, San Jose, CA 95120  
hazan@us.ibm.com

**Satyen Kale**

Microsoft Research

1 Microsoft Way, Redmond, WA 98052  
sakale@microsoft.com

## Abstract

Prediction from expert advice is a fundamental problem in machine learning. A major pillar of the field is the existence of learning algorithms whose average loss approaches that of the best expert in hindsight (in other words, whose average regret approaches zero). Traditionally the regret of online algorithms was bounded in terms of the number of prediction rounds.

Cesa-Bianchi, Mansour and Stoltz [4] posed the question whether it is possible to bound the regret of an online algorithm by the *variation* of the observed costs. In this paper we resolve this question, and prove such bounds in the fully adversarial setting, in two important online learning scenarios: prediction from expert advice, and online linear optimization.

## 1 Introduction

A cornerstone of modern machine learning are algorithms for prediction from expert advice. The seminal work of Littlestone and Warmuth [12], Vovk [13] and Freund and Schapire [6] gave algorithms which, under fully adversarial cost sequences, attain average cost approaching that of the best expert in hindsight.

To be more precise, consider a prediction setting in which an online learner has access to  $n$  experts. Iteratively, the learner may choose the advice of any expert deterministically or randomly. After choosing a course of action, an adversary reveals the cost of following the advice of the different experts, from which the expected cost of the online learner is derived. The classic results mentioned above give algorithms which sequentially produce randomized decisions, such that the difference between the (expected) cost of the algorithm and the best expert in hindsight grows like  $O(\sqrt{T} \log n)$ , where  $T$  is the number of prediction iterations. This extra additive cost is known as the regret of the online learning algorithm.

However, *a priori* it is not clear why online learning algorithms should have high regret (growing with the number of iterations) in an unchanging environment. As an extreme example, consider a setting in which there are only two experts. Suppose that the first expert always incurs cost 1, whereas

the second expert always incurs cost  $\frac{1}{2}$ . One would expect to “figure out” this pattern quickly, and focus on the second expert, thus incurring a total cost that is at most  $\frac{T}{2}$  plus at most a constant extra cost (irrespective of the number of rounds  $T$ ), thus having only constant regret. However, any straightforward application of previously known analyses of expert learning algorithms only gives a regret bound of  $\Theta(\sqrt{T})$  in this simple case (or very simple variations of it).

More generally, the natural bound on the regret of a “good” learning algorithm should depend on *variation* in the sequence of costs, rather than purely on the number of iterations. If the cost sequence has low variation, we expect our algorithm to be able to perform better.

This intuition has a direct analog in the stochastic setting: here, the sequence of experts’ costs are independently sampled from a distribution. In this situation, a natural bound on the rate of convergence to the optimal expert is controlled by the variance of the distribution (low variance should imply faster convergence). This was formalized by Cesa-Bianchi, Mansour and Stoltz [4], who assert that “*proving such a rate in the fully adversarial setting would be a fundamental result*”.

In this paper we prove the first such regret bounds on online learning algorithms in two important scenarios: prediction from expert advice, and the more general framework of online linear optimization. Our algorithms have regret bounded by the variation of the cost sequence, in a manner that is made precise in the following sections. Thus, our bounds are tighter than *all* previous bounds, and hence yield better bounds on the applications of previous bounds (see, for example, the applications in [4]).

### 1.1 Online linear optimization

Online linear optimization [10] is a general framework for online learning which has received much attention recently. In this framework the decision set is an arbitrary bounded, closed, convex set in Euclidean space  $K \subseteq \mathbb{R}^n$  rather than a fixed set of experts, and the costs are determined by adversarially constructed vectors,  $f_1, f_2, \dots \in \mathbb{R}^n$ , such that the cost of point  $x \in K$  is given by  $f_t \cdot x$ . The online learner iteratively chooses a point in the convex set  $x_t \in K$ , and then the cost vector  $f_t$  is revealed and the cost  $f_t \cdot x_t$  is incurred. The performance of online learning algorithms is measured by the regret, which is defined as the difference in the total cost of the sequence of points chosen by the algorithm, viz.

$\sum_{t=1}^T f_t \cdot x_t$ , and the total cost of the least cost fixed point in hindsight, viz.  $\min_{x \in K} \sum_{t=1}^T f_t \cdot x$ .

Several decision problems fit very naturally in this framework. For example, in the online shortest path problem the online learner has to repeatedly choose a path in a given graph from a source node to a destination node. Her cost is the total length of the path according to weights which are chosen by an adversary. This problem can be cast as an online linear optimization problem, where the decision space is the set of all distributions over paths in the graph connecting the source to the destination. Even though this set sits in exponential dimensional Euclidean space, by thinking of a distribution over paths as a flow in the graph, it is possible to efficiently represent the decision space as a polytope in  $\mathbb{R}^{|E|}$  ( $E$  denotes the set of edges in the given graph), described by  $O(|E|)$  constraints, and translate the cost functions to this new domain as well.

The general online linear optimization framework allows for efficient and natural algorithms based on the gradient descent update rule coupled with Euclidean projections [8, 14]. Specifically, we consider Zinkevich’s Lazy Projection algorithm [14]. This algorithm runs online gradient descent on an auxiliary sequence of points and chooses the projections of these auxiliary points on the convex set in every iteration. This algorithm was shown to have regret  $O(\sqrt{T})$ .

The crucial geometric intuition which allows us to prove regret bounds based on the variation of the cost sequence can be summarized by the following intuitive fact: the distance between successive projections for the Lazy Projection algorithm is directly related to the variation of the cost sequence.

We now describe our bounds. Define the variation of the sequence of cost functions to be  $\text{VAR}_T = \sum_{t=1}^T \|f_t - \mu_T^*\|^2$ , where  $\mu_T^* = \frac{1}{T} \sum_{t=1}^T f_t$  is the mean of the sequence. Our analysis of the Lazy Projection algorithm yields the following regret bound:

$$\text{Regret} \leq O(\sqrt{\text{VAR}_T}).$$

## 1.2 Prediction from expert advice

Prediction from expert advice can be cast as a special case of online linear optimization: the decision space is the simplex of all distributions on  $n$  experts. The expectation operator provides a linear cost function on the simplex via the costs of the experts. Hence, our result for online linear optimization already implies variation bounds for regret in the case of prediction from expert advice.

However, this bound is suboptimal, as it depends on the variation of all experts rather than, say, the maximum variation of a single expert. This issue is familiar to learning theorists: “Euclidean algorithms” such as gradient descent attain performance which relates to the Euclidean norm of the cost functions (or variations in our case). While this Euclidean flavor is optimal in certain cases (i.e. when the underlying convex set is the hyper-cube), for certain convex bodies such as the simplex, better performance can be achieved. The multiplicative update algorithms such as EG [11] and FPL\* [10] attain regret which is proportional to  $O(R\sqrt{T \log n})$  where  $R$  is a bound on the  $\ell_\infty$  norm of the cost functions.

By analogy with the online linear optimization case, for a sequence of cost vectors  $f_1, f_2, \dots, f_T \in \mathbb{R}^n$ , where  $f_t(i)$

is the cost of expert  $i$  in the  $t^{\text{th}}$  round, we would expect to be able to bound the regret of online linear optimization over the simplex by something like  $O(\sqrt{\text{VAR}_T^\infty \log n})$ , where

$$\text{VAR}_T^\infty = \max_{i \in n} \left\{ \sum_{t=1}^T |f_t(i) - \mu_T^*(i)|^2 \right\}$$

is the maximum variation in costs amongst the different experts (as before,  $\mu_T^*(i) = \frac{1}{T} \sum_{t=1}^T f_t(i)$  is the mean cost of the  $i^{\text{th}}$  expert). In fact, our bound is even a bit stronger,

$$\text{Regret}(T) = O\left(\sqrt{\text{VAR}_T^{\max} \log n}\right).$$

Here  $\text{VAR}_T^{\max} \leq \text{VAR}_T^\infty$ , and is defined to be

$$\text{VAR}_T^{\max} = \max_{t \leq T} \{\text{VAR}_t(\ell_t)\},$$

where  $\text{VAR}_t(i)$  is the variation in costs of expert  $i$  up to the  $t^{\text{th}}$  round, and  $\ell_t$  is the best expert till the  $t^{\text{th}}$  round.

Whereas for the general online linear optimization we consider the well-known Lazy Projection algorithms and our results are novel by tighter analysis, for the case of prediction from expert advice we need to consider a new algorithm. We can prove that existing variants of the multiplicative weights algorithms do not attain the performance above, and instead consider a different variant of update rule, in which the distribution at time  $t$ , denoted  $x_t$  is defined to be

$$x_t(i) \propto \exp\left(-\eta \sum_{\tau=1}^{t-1} f_\tau(i) - 4\eta^2 \sum_{\tau=1}^{t-1} (f_\tau(i) - \mu_\tau(i))^2\right),$$

where  $\eta$  is a learning rate parameter and  $\mu_t = \frac{1}{t} \sum_{\tau=1}^{t-1} f_\tau$  is the (approximate) mean cost vector at iteration  $t$ . That is, *the distribution over experts explicitly takes into account the variation in their costs*. As far as we know this is a new variant of the multiplicative update algorithms family, and it is necessary to include this feature to prove variation bounds on the regret.

## 1.3 Discussion of the results

Cesa-Bianchi, Mansour and Stoltz [4] discussed desiderata for *fundamental* regret bounds for the expert prediction problem: invariance under translation and rescaling of costs vectors. Invariance under translation implies that the bounds depend only on the effective ranges of the cost vectors in each round, rather than the absolute ranges (by effective range, we mean the maximum difference between the costs in any given round). This is because if any round, the cost vectors are all changed by the same amount, the difference between the expected cost of the algorithm in that round and the cost of any given expert remains the same as without the translation. Our regret bounds enjoy this translation invariance property: this is a direct consequence of the variation bound. This implies, for instance, that it doesn’t matter what sign the costs are, and in fact our bounds are robust enough to handle mixed signs in costs.

Rescaling invariance implies that the bound continues to hold even if all the cost vectors are scaled by the same factor. Again, our regret bounds enjoy rescaling invariance since the regret and the square-root variation scale by the same factors.

We make crucial use of these invariance properties in our analysis; the invariance allows us to normalize the cost vectors in ways that make them easier to reason about.

## 1.4 The stationary stochastic setting vs. an adversary

A point made by [4] is that the variation bounds on the regret essentially match the performance of a natural algorithm in the stochastic setting in which the payoffs are generated by a stationary stochastic process. Let us give a rough sketch of why this is true. Consider a setting of online linear optimization over the unit ball. Suppose that the cost functions are generated by a stationary stochastic process, such that in each iteration the cost function is independently sampled from a fixed distribution with some mean vector  $\mu$ . For a long enough sequence of cost functions drawn from this distribution, the best point in hindsight is essentially the least cost point with respect to the cost vector  $\mu$ .

Let  $\bar{\mu}$  be the observed mean of samples. The natural algorithm uses  $\bar{\mu}$  as proxy for the actual mean and chooses its point with  $\bar{\mu}$  as a cost vector, and this can be shown to be optimal. It is a standard fact that the variance of  $\bar{\mu}$  decreases inversely with the number of samples. Thus, if  $\sigma^2$  is the variance of the distribution, then the variance of  $\bar{\mu}$  after  $t$  iterations is  $\frac{\sigma^2}{t}$ . The expected regret on iteration  $t$  is proportional to the standard deviation  $\frac{\sigma}{\sqrt{t}}$ , and thus the total regret of the optimal predictor is on the order of  $\sum_{t=1}^T \frac{\sigma}{\sqrt{t}} = O(\sqrt{\sigma^2 T}) = O(\sqrt{\text{VAR}_T})$ .

Hence, the optimal achievable regret in this simple setting is proportional to square root of the total variation. In the sequel we prove that the same regret (up to constant factors) can be achieved in the fully adversarial setting, i.e. in a setting in which the cost functions are chosen completely adversarially. In the stationary stochastic setting, the average cost converges to the average optimum cost at a speed that depends on the variance of the distribution: lower variance implies faster convergence. Hence, by proving the variation bounds on the regret, we give strong indication that *online linear optimization in the adversarial setting is as efficient as in the stationary stochastic setting*.

## 1.5 A brief history of prediction

It is incredible that as early as the late fifties, Hannan [7] devised an efficient algorithm for online decision making. Hannan's algorithm proceeds by adding a perturbation to the costs of actions seen so far, and choosing the action with least cost (taking into account the perturbations). He proves that the regret of an online player using his algorithm grows like  $O(\sqrt{T})$  where  $T$  is the number of prediction iterations.

Since then much water has flown under the bridge and many experts have predicted: this includes the aforementioned influential multiplicative update family of algorithms [12, 13, 6], Cover's universal portfolio prediction problem [5] and the extensions of Follow-The-Perturbed-Leader [10] to online optimization and complex decision problems such as online shortest paths. The machine learning community has extended these fundamental results into a beautiful theory of general prediction using Bregman divergences and generalized projections (in order to do justice to the numerous contributors we refer to the credits in the comprehensive book of [3]). This work refined upon the basic regret bound of  $O(\sqrt{T})$ . This refinement, however, deals with the constants multiplying the  $\sqrt{T}$  term.

Freund and Schapire [6] showed that a Multiplicative Weights algorithm based on the Weighted Majority algorithm attains regret bounds of  $O\left(\sqrt{R \sum_{t=1}^T f_t(i^*) \log n}\right)$ , where it is assumed that all costs are in the range  $[0, R]$ , and  $i^*$  is the best expert in hindsight. In the case when the costs lie in the range  $[-R, R]$ , Allenberg-Neeman and Neeman [1] showed that there is an expert  $i$  such that the regret can be bounded by  $O\left(\sqrt{R \sum_{t=1}^T |f_t(i)| \log n}\right)$ . Most recently Cesa-Bianchi, Mansour and Stoltz [4] gave the first second-order regret bounds: they proved a bound of  $O\left(\sqrt{A_T^{\max} \log n}\right)$  where  $A_T^{\max} = \max_{t \leq T} \left\{ \sum_{\tau=1}^t f_\tau(\ell_t)^2 \right\}$  is the maximum, over all the time periods  $t$ , of the sum of squares of losses up to time  $t$  of the best expert at time  $t$ . They suggest, and indeed as we argue in the previous section it makes intuitive sense, that the it should be possible to get a bound that scales as  $\sqrt{\text{VAR}_T^{\max}}$ .

In this paper we prove their conjecture to be correct, in effect providing the optimal regret bounds up to constant factors.

## 2 Notation and background

The following definitions and derivations may be familiar to experts in learning theory, who may wish to proceed directly to the next section.

In the online linear optimization problem, the decision space is a closed, bounded, convex set  $K \in \mathbb{R}^n$ , and we are sequentially given a series of linear cost functions  $f_t : K \rightarrow \mathbb{R}$  for  $t = 1, 2, \dots$ . With some abuse of notation, we also write the functions as  $f_t(x) = f_t \cdot x$  for some vector  $f_t \in \mathbb{R}^n$ .

The algorithm iteratively produces a point  $x_t \in K$  in every round  $t$ , without knowledge of  $f_t$  (but using the past sequence of cost functions), and incurs the cost  $f_t(x_t)$ . The regret at time  $T$  is defined to be

$$\text{Regret}(f_1, f_2, \dots, f_T) := \sum_{t=1}^T f_t(x_t) - \min_{x \in K} \sum_{t=1}^T f_t(x).$$

Usually, we will drop the cost vectors from the regret notation when they are clear from context. For convenience, we define  $f_0 = 0$ , and let  $F_t = \sum_{\tau=0}^{t-1} f_\tau$ .

We proceed to describe a widely used algorithmic technique in online learning, on the basis of which we will derive our algorithms.

Since our goal is to get regret bounded by the variation in the cost sequence, intuitively, a Follow-The-Leader (FTL) type algorithm, which always chooses the best point so far to use in the next round, should perform well if the variation is low. The FTL algorithm by itself doesn't usually guarantee low regret, mainly because it is inherently unstable: it may swing wildly from one point to another from one iteration to the next at very little provocation (for example, consider the case of expert prediction with 2 experts for the following sequence of cost vectors:  $(1/2, 0), (0, 1), (1, 0), (0, 1), \dots$ ). To make it stable, we add a strictly convex regularization function  $R(x)$  before computing the leader. The generic algorithm which results is shown below, and is called Follow The Regularized Leader (FTRL):

---

**Algorithm 1** FTRL

---

- 1: Let  $K$  be a convex set
- 2: Input: parameter  $\eta > 0$ , regularization function  $R(x)$ .
- 3: **for**  $t = 1$  to  $T$  **do**
- 4:   Use  $x_t \triangleq \min_{x \in K} \left( F_t \cdot x + \frac{1}{\eta} R(x) \right)$ ,
- 5:   Receive  $f_t$
- 6: **end for**

---

A crucial observation regarding the FTRL algorithm which we use in the analysis is its equivalence to the following algorithm, which we call Follow the Lazy Projected Leader (FLPL). This algorithm maintains an auxiliary sequence of points which are updated using a gradient descent type algorithm, which are then projected into the convex set using the Bregman divergence  $B^R$  defined by  $R$ :

$$B^R(x, y) = R(x) - R(y) - \nabla R(y) \cdot (x - y).$$

The algorithm as it is given has an implicit update, whose implementation we ignore for now (in this paper we are only concerned with the Euclidean and Relative Entropy divergences, in which case the updates are efficient).

---

**Algorithm 2** FLPL

---

- 1: Let  $K$  be a convex set
- 2: Input: parameter  $\eta > 0$ , regularizer function  $R(x)$ .
- 3: **for**  $t = 1$  to  $T$  **do**
- 4:   If  $t = 1$ , choose  $y_1$  such that  $\nabla R(y_1) = 0$ .
- 5:   If  $t > 1$ , choose  $y_t$  such that  $\nabla R(y_t) = \nabla R(y_{t-1}) - \eta f_{t-1}$ .
- 6:   Project according to  $B^R$ :

$$x_t = \arg \min_{x \in K} B^R(x, y_t)$$

- 7: **end for**

---

In fact, the two algorithms above are identical. This is perhaps not surprising, given what is known about the so called “mirror-descent” algorithm (e.g. [3]). Nevertheless this fact is crucial for our later derivations, and we did not find this precise statement elsewhere, hence we include a short proof.

**Lemma 1** *The two algorithms above produce identical predictions, i.e.*

$$\arg \min_{x \in K} \left( F_t \cdot x + \frac{1}{\eta} R(x) \right) = \arg \min_{x \in K} B^R(x, y_t).$$

**Proof:** First, let us observe that the unconstrained optimum  $x^* = \arg \min_{x \in \mathbb{R}^n} \left( F_t \cdot x + \frac{1}{\eta} R(x) \right)$  satisfies

$$F_t + \frac{1}{\eta} \nabla R(x^*) = 0$$

By induction, the above equation is also satisfied for  $y_t$ . Since  $R(x)$  is assumed to be strictly convex, there is only one solution for the above equation and thus  $y_t = x^*$ . Hence,

$$\begin{aligned} B_R(x, y_t) &= R(x) - R(y_t) - \nabla R(y_t) \cdot (x - y_t) \\ &= R(x) - R(y_t) + \eta F_t \cdot (x - y_t). \end{aligned}$$

Since  $R(y_t)$  and  $F_t \cdot y_t$  are constants (i.e. independent of  $x$ ),  $B_R(x, y_t)$  is minimized at the point  $x$  that minimizes  $R(x) + \eta F_t \cdot x$ , which implies that

$$\arg \min_{x \in K} B_R(x, y_t) = \arg \min_{x \in K} \left( F_t \cdot x + \frac{1}{\eta} R(x) \right).$$

■

One important property which follows from the first characterization of  $x_t$  is the following standard bound on the regret, due to Kalai and Vempala [10], called the Follow-The-Leader/Be-The-Leader (FTL-BTL) inequality:

**Lemma 2** *The regret of the FTRL (or equivalently, the FLPL) algorithm is bounded as:*

$$\text{Regret} \leq \sum_{t=1}^T f_t \cdot (x_t - x_{t+1}) + \frac{1}{\eta} [R(x_T) - R(x_0)].$$

### 3 Algorithms and main results

In this section we describe the algorithms for which we prove variation bounds, and state formally their performance guarantees.

#### 3.1 Online linear optimization

We start by describing our result for online linear optimization. Following the notation defined in the previous section, we assume that  $K \subseteq \mathbb{B}_n$ , where  $\mathbb{B}_n$  is the unit ball in  $\mathbb{R}^n$ , and that  $0 \in K$ . This is without loss of generality, and can be assumed by a suitable scaling and translation of  $K$ . Scaling  $K$  down by its diameter  $D$  makes the diameter 1 and scales the regret down by  $D$  as well, and changing the coordinate system so that  $K$  contains the origin doesn’t change the regret bound. Here, we are making use of the translation invariance of our regret bounds.

We also assume that for all  $t$ ,  $\|f_t\| \leq 1$ . If we have some other bound  $R$  on  $\|f_t\|$ , then we scale down the  $f_t$ ’s by  $R$  to get new cost vectors  $f'_t$  such that  $\|f'_t\| \leq 1$ . We can then run the algorithm pretending as if  $f'_t$  is the sequence of cost vectors.

Define the variation of sequence of cost vectors  $f_1, \dots, f_T$  to be

$$\text{VAR}_T(f_1, f_2, \dots, f_T) = \sum_{t=1}^T \|f_t - \mu\|^2,$$

where  $\mu = \frac{1}{T} \sum_{t=1}^T f_t$  is the vector that minimizes the above expression. Usually, we will drop the cost vectors from the notation for the variation, refer to it simply as  $\text{VAR}_T$ , when the cost vectors are clear from context. To see that scaling has no effect on the regret bound, note

$$\text{VAR}_T(f'_1, \dots, f'_T) = \frac{1}{R^2} \text{VAR}_T(f_1, \dots, f_T),$$

and

$$\text{Regret}(f'_1, \dots, f'_T) = \frac{1}{R} \text{Regret}(f_1, \dots, f_T).$$

Thus, if  $\text{Regret}(f'_1, \dots, f'_T) = O(\sqrt{\text{VAR}_T(f'_1, \dots, f'_T)})$ , then  $\text{Regret}(f_1, \dots, f_T) = O(\sqrt{\text{VAR}_T(f_1, \dots, f_T)})$ . This is exactly the rescaling invariance discussed earlier.

For ease of notation, we define  $f_0 = 0$ , and for any  $t > 0$ , let  $F_t = \sum_{\tau=0}^{t-1} f_\tau$  and  $\mu_t = \frac{1}{t} F_t = \frac{1}{t} \sum_{\tau=0}^{t-1} f_\tau$ . We instantiate the FTRL/FLPL algorithm with the regularization function  $R(x) = \frac{1}{2} \|x\|^2$ . This regularization was considered many times before, and the only change hereby is to choose a different ‘‘learning rate’’  $\eta$ , which will enable us to prove the novel regret bounds. Since  $\nabla R(x) = x$  for this regularization, the algorithm that results is:

---

**Algorithm 3** Lazy Projection

---

- 1: Let  $K$  be a convex set
  - 2: Input: parameter  $\eta > 0$ .
  - 3: **for**  $t = 1$  to  $T$  **do**
  - 4:   If  $t = 1$ , choose  $y_1 = 0$ .
  - 5:   If  $t > 1$ , let  $y_t = y_{t-1} - \eta f_{t-1}$ .
  - 6:   Use  $x_t = \arg \min_{x \in K} \|x - y_t\|$ .
  - 7: **end for**
- 

Our main theorem with respect to online linear optimization is:

**Theorem 3** *Let  $f_t$ , for  $t = 1, 2, \dots, T$ , be a sequence of cost vectors to the experts so that  $\|f_t\| \leq 1$ . Setting  $\eta = \min\{2/\sqrt{\text{VAR}_T}, 1/6\}$ , the regret of the Lazy Projection algorithm is bounded by*

$$\text{Regret} \leq 16\sqrt{\text{VAR}_T}.$$

Of course, an upper bound on the total variation  $\text{VAR}_T$  may not be known in advance. Even so, standard  $\eta$ -halving tricks (start with  $\eta = 1/6$ , and halve  $\eta$  as soon as the variation quadruples, and restart the algorithm) immediately give an  $O(\sqrt{\text{VAR}_T})$  regret bound. The formal application of this standard trick is omitted from this extended abstract.

### 3.2 Prediction from expert advice

In the expert learning problem, we assume that we have access to  $n$  experts. In each round  $t$ , we choose a distribution  $x_t$  over the experts and choose an expert from it. Then, we obtain a cost vector  $f_t$  which specifies a cost  $f_t(i)$  for every expert, and we incur the cost of the chosen expert. Our goal is to bound the total expected cost of the algorithm (i.e.  $\sum_{t=1}^T f_t \cdot x_t$ ) relative to the total cost of the expert with minimum total cost in hindsight (i.e.  $\min_i \sum_{t=1}^T f_t(i)$ ).

For simplicity, we assume that all costs  $f_t(i) \in [0, 1]$ . This can be assumed without loss of generality from the rescaling and translation invariance of our final regret bounds. In general, all we need is bound  $R$  on the maximum value of  $f_t(i) - f_t(j)$  over all rounds and all pairs of experts  $i, j$ . In each round, the algorithm can be run by scaling the costs of all experts down by  $R$ , and then subtracting out the minimum cost in each round. As in the case of online linear optimization, this scaling and translation doesn’t affect the square-root variation bound on the regret.

As mentioned before, this setting is a special case of the online linear optimization where the domain  $K$  is the simplex (denoted  $\Delta$ ) of distributions over the experts. To design an algorithm for this special case, we need a different regularization function,  $\text{ne}(x) = \sum_i x_i \ln x_i - x_i$ . The Bregman

divergence which arises from this is the un-normalized relative entropy (c.f Herbster and Warmuth [9]), defined on  $\mathbb{R}_+^n$ , called as follows:

$$D_{\text{ne}}(x, y) := \sum_i y_i \cdot \ln \frac{y_i}{x_i} + y_i - x_i.$$

Note that when  $x, y \in \Delta$ ,  $D_{\text{ne}}(x, y)$  is the relative entropy between  $x$  and  $y$ , and  $\text{ne}(x)$  is the negative entropy of  $x$ . The Bregman projection on the simplex with the un-normalized relative entropy divergence is implemented simply by scaling all the coordinates so that they sum to 1.

A significant twist on the usual multiplicative weights algorithm is that we modify the cost functions to explicitly take into account the variation: we actually run the FTRL/FLPL algorithm on the sequence of cost vectors  $\tilde{f}_1, \tilde{f}_2, \dots$  where

$$\tilde{f}_t(i) = [f_t(i) + 4\eta(f_t(i) - \mu_t(i))^2],$$

where  $\mu_t = \frac{1}{t} \sum_{\tau=0}^{t-1} f_\tau$ . As before, we use the convention that  $f_0 = 0$ .

For ease of notation, for a vector  $x$ , we define the vector  $x^2$  as  $x^2(i) = x(i)^2$ . Thus, we can write  $\tilde{f}_t$  compactly as  $\tilde{f}_t = f_t + 4\eta(f_t - \mu_t)^2$ . The algorithm which results is given below:

---

**Algorithm 4** Variation MW

---

- 1: Input: parameter  $\eta > 0$ .
  - 2: **for**  $t = 1$  to  $T$  **do**
  - 3:   If  $t = 1$ , choose  $y_1 = \vec{1}$ , the all 1’s vector.
  - 4:   If  $t > 1$ , let  $y_t(i) = y_{t-1}(i) \exp(-\eta \tilde{f}_{t-1}(i))$ , where  $\tilde{f}_{t-1}(i) = f_{t-1}(i) + 4\eta(f_{t-1}(i) - \mu_{t-1}(i))^2$ .
  - 5:   Use  $x_t = y_t/Z_t$ , where  $Z_t = \sum_i y_t(i)$ .
  - 6: **end for**
- 

Define

$$\text{VAR}_T^{\max} = \max_{t \leq T} \{\text{VAR}_t(\ell_t)\},$$

where  $\ell_t$  is the best expert till the  $t^{\text{th}}$  round, and  $\text{VAR}_t(i) = \sum_{\tau=1}^t (f_\tau(i) - \mu_t^*(i))^2$  where  $\mu_t^*(i) = \frac{1}{t} \sum_{\tau=1}^t f_\tau(i)$  is the mean cost of the  $i^{\text{th}}$  expert till the  $t^{\text{th}}$  round. Our main result concerning prediction from expert advice is

**Theorem 4** *Let  $f_t$ , for  $t = 1, 2, \dots, T$ , be a sequence of cost vectors to the experts so that  $f_t(i) \in [0, 1]$ . Setting  $\eta = \min \left\{ \sqrt{\log(n)/4\text{VAR}_T^{\max}}, 1/10 \right\}$ , the regret of the Variation MW algorithm is bounded by*

$$\text{Regret} \leq 8\sqrt{\text{VAR}_T^{\max} \log(n)} + 10 \log(n).$$

Again,  $\eta$ -halving tricks can be used to obtain this result in the case when  $\text{VAR}_T^{\max}$  is not known ahead of time. The additive  $\log(n)$  term is inherent in all expert learning algorithms and also appears in all previously known regret bounds.

## 4 Analysis of the Lazy Projection algorithm

In this section we prove Theorem 3. The proof uses the dual characterization of the FTRL type algorithms introduced previously: on one hand we follow the standard methodology of the Follow-The-Leader type algorithms, bounding the regret by distance between consecutive predictions. On the other hand we use the fact that these predictions are projections of aggregate cost functions, and analyze the distance between successive projections. In fact, this latter analysis is the main crux of the proof - we refine previous approaches by giving a tighter bound on this distance which is based on simple geometrical intuition.

### Proof: (Theorem 3)

In order to aid understanding, we present the proof as a series of lemmas. We defer the proofs of the lemmas to after the present proof. We start by invoking the FTL-BTL inequality (Lemma 2) to obtain the following bound:

#### Lemma 5

$$\text{Regret} \leq \sum_{t=1}^T (f_t - \mu_t) \cdot (x_t - x_{t+1}) + \frac{1}{\eta}.$$

We proceed to relate the distance between successive projections to the variation in the cost vectors. This lemma is the main crux of the proof, and is based on the geometric intuition depicted in Figure 1. The idea in the proof is that if the sequence of cost vectors has low variation, then the cumulative cost vector  $F_t$  is far away from the convex body, and in such a case, the distance between successive projections can be bounded in terms of the length of the component of  $f_t$  orthogonal to  $F_t$ , which can in turn be bounded in terms of  $\|f_t - \mu_t\|$ , since  $\mu_t = \frac{1}{t}F_t$ .

#### Lemma 6

For all  $t$ , we have:

$$\|x_t - x_{t+1}\| \leq \frac{3\eta}{2} \|f_t - \mu_t\| + \frac{2}{t}.$$

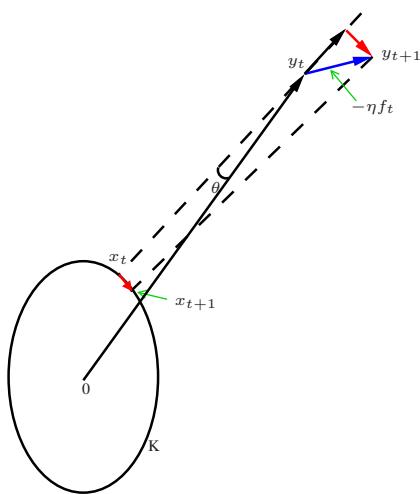


Figure 1: The distance between successive projections, viz.  $\|x_t - x_{t+1}\|$ , is bounded by the length of the component of  $-\eta f_t$  orthogonal to the  $y_t - x_t$ .

For ease of notation, we define a parameter of the cost vectors which will be further used in the analysis:

$$\rho(T) := \sum_{t=1}^T \frac{1}{t} \|f_t - \mu_t\|.$$

This parameter measures the variation of the cost vectors. Using the Cauchy-Schwartz inequality and Lemma 6 we get

$$\begin{aligned} & (f_t - \mu_t) \cdot (x_t - x_{t+1}) \\ & \leq \|f_t - \mu_t\| \cdot \left[ \frac{3\eta}{2} \|f_t - \mu_t\| + \frac{2}{t} \right] \\ & \leq \frac{3\eta}{2} \|f_t - \mu_t\|^2 + \frac{2\|f_t - \mu_t\|}{t}. \end{aligned}$$

Plugging this into the regret bound of Lemma 5 gives us the following bound:

$$\text{Regret} \leq \frac{3\eta}{2} \sum_{t=1}^T \|f_t - \mu_t\|^2 + 2\rho(T) + \frac{1}{\eta}. \quad (1)$$

To proceed from here, we use the following Lemma (which, curiously enough, is proved using the analysis of an online learning algorithm that has nothing to do with the present setting!):

#### Lemma 7

For any vector  $\mu$ , we have:

$$\sum_{t=1}^T \|f_t - \mu_t\|^2 \leq \sum_{t=1}^T \|f_t - \mu\|^2 + 4\rho(T).$$

Plugging into equation (1) we get that, for any vector  $\mu$  (and in particular, for  $\mu = \mu_T^* := \frac{1}{T} \sum_{t=1}^T f_t$ ),

$$\text{Regret} \leq \frac{3\eta}{2} \sum_{t=1}^T \|f_t - \mu\|^2 + (2 + 6\eta)\rho(T) + \frac{1}{\eta}.$$

We can bound on  $\rho(T)$  as follows:

#### Lemma 8

For any vector  $\mu$ , we have:

$$\rho(T) \leq 3 \sqrt{\sum_{t=1}^T \|f_t - \mu\|^2}.$$

Finally, by setting  $\eta = \min\{2/\sqrt{\text{VAR}_T}, 1/6\}$ , the proof is complete. ■

We now give the omitted proofs of Lemmas used in the above proof.

### Proof: (Lemma 5)

By definition of  $x_t$ , we know that

$$F_t \cdot x_t + \frac{1}{2\eta} \|x_t\|^2 \leq F_t \cdot x_{t+1} + \frac{1}{2\eta} \|x_{t+1}\|^2.$$

Recall that  $\mu_t = F_t/t$ . Hence,

$$\begin{aligned} \sum_{t=1}^T \mu_t \cdot (x_t - x_{t+1}) &= \sum_{t=1}^T \frac{F_t}{t} \cdot (x_t - x_{t+1}) \\ &\leq \sum_{t=1}^T \frac{1}{t} \cdot \frac{1}{2\eta} (\|x_{t+1}\|^2 - \|x_t\|^2) \\ &\leq \frac{1}{2\eta} \sum_{t=2}^T \|x_t\|^2 \cdot \left( \frac{1}{t-1} - \frac{1}{t} \right) + \frac{\|x_{T+1}\|^2}{2\eta T} \\ &\leq \frac{1}{2\eta}. \end{aligned}$$

Here, we use the fact that  $\|x_t\| \leq 1$ . The stated bound then follows from Lemma 2.  $\blacksquare$

### Proof: (Lemma 6)

We split up the analysis in two cases:

1.  $\|F_t\| \leq 2/\eta$ : Assume that  $\|F_t\| > 0$ . Since  $x_t$  and  $x_{t+1}$  are the projections of  $y_t$  and  $y_{t+1}$  respectively on  $K$ , by the Projection Lemma 9 we have

$$\begin{aligned} \|x_t - x_{t+1}\| &\leq \|y_t - y_{t+1}\| \\ &= \eta \|f_t\| \\ &\leq \eta \|f_t - \mu_t\| + \eta \|\mu_t\| \\ &\leq \eta \|f_t - \mu_t\| + \frac{2\|\mu_t\|}{\|F_t\|} \\ &= \eta \|f_t - \mu_t\| + \frac{2}{t}. \end{aligned}$$

If  $\|F_t\| = 0$ , then the Projection Lemma 9 implies that  $\|x_t - x_{t+1}\| \leq \eta \|f_t\| = \eta \|f_t - \mu_t\|$ , so the stated bound still holds.

2.  $\|F_t\| \geq 2/\eta$ : we first show the following bound:

$$\|x_t - x_{t+1}\| \leq \eta \|f_t - \mu_t\| + \|f_t\|/\|F_t\|. \quad (2)$$

Consider two unit vectors:  $u$  in the direction  $y_t - x_t$ , and  $v$  in the direction  $y_t$ . We claim that the sine of the angle  $\theta$  between these vectors is at most  $1/\eta\|F_t\|$ . To see this, consider the triangle formed by the points  $0, x_t, y_t$ . We are interested in the angle  $\theta$  at vertex  $y_t$  (see Figure 1). Let  $\vartheta$  be the angle at  $x_t$ . By the law of sines, we have

$$\sin(\theta) = \frac{\|x_t\| \sin(\vartheta)}{\|y_t\|} \leq \frac{1}{\|y_t\|} = \frac{1}{\eta\|F_t\|},$$

where the inequality follows because  $\|x_t\| \leq 1$  and  $\sin(\vartheta) \leq 1$ .

Now, we consider the components of  $f_t$  along  $u$  and  $v$ : define  $f_t^u = (f_t \cdot u)u$  and  $f_t^v = (f_t \cdot v)v$ . Consider the point  $y_t - \eta f_t^u$ . Since it lies on the line joining  $y_t$  to  $x_t$ , its projection on  $K$  is also  $x_t$ . Here, we use the fact that  $y_t - \eta f_t^u$  is outside  $K$ : this is because

$$\|y_t - \eta f_t^u\| \geq \|y_t\| - \eta \|f_t^u\| \geq \eta \|F_t\| - \eta \geq 1.$$

By the Projection Lemma 9, we have

$$\|x_{t+1} - x_t\| \leq \|y_{t+1} - (y_t - \eta f_t^u)\| = \eta \|f_t - f_t^u\|. \quad (3)$$

Let  $x$  be the projection of  $f_t^v$  on the subspace spanned by  $u$  (i.e.  $x = (f_t^v \cdot u)u$ ). Then, since  $f_t^u$  is the projection of  $f_t$  in the subspace spanned by  $u$ , it is the closest point to  $f_t$  in the subspace, and since  $x$  is also in the subspace, we have

$$\begin{aligned} \|f_t - f_t^u\| &\leq \|f_t - x\| \\ &\leq \|f_t - f_t^v\| + \|f_t^v - x\| \\ &= \|f_t - f_t^v\| + \|f_t^v\| \sin(\theta) \\ &\leq \|f_t - f_t^v\| + \|f_t\|/\eta\|F_t\| \\ &\leq \|f_t - \mu_t\| + \|f_t\|/\eta\|F_t\|. \end{aligned}$$

The last inequality follows because  $f_t^v$  is the closest point to  $f_t$  in the subspace spanned by  $v$ , and  $\mu_t$  is a point in this subspace. Plugging this bound into (3), we get (2).

Now, we have the following bound on  $\|f_t\|/\|F_t\|$ :

$$\frac{\|f_t\|}{\|F_t\|} \leq \frac{\|f_t - \mu_t\| + \|\mu_t\|}{\|F_t\|} \leq \frac{\eta\|f_t - \mu_t\| + \frac{1}{t}}{\|F_t\|}. \quad (4)$$

Plugging (4) into (2), we get the required bound.  $\blacksquare$

### Proof: (Lemma 7)

We may assume that  $\|\mu\| \leq 1$ , since the right hand side is minimized at  $\mu = \frac{1}{T} \sum_{t=1}^T f_t$ . The statement of the lemma is essentially bounding the regret of the FTL algorithm played on the sequence of cost functions  $c_t(x) = \|x - f_t\|^2$ , for  $t = 0, 1, 2, \dots, T$ , with the convex domain the unit ball  $\mathbb{B}_n$ . This is because the leader in round  $t$  is

$$\arg \min_{x \in \mathbb{B}_n} \left\{ \sum_{\tau=0}^{t-1} \|x - f_\tau\|^2 \right\} = \frac{1}{t} \sum_{\tau=0}^{t-1} f_\tau = \mu_t.$$

We assume here that the first point played by the algorithm is 0. Then by the FTL-BTL inequality (Lemma 2), the regret of the FTL algorithm can be bounded as (here, the regularization function  $R(x)$  is null):

$$\begin{aligned} \text{Regret} &\leq c_0(0) - c_0(\mu_1) + \sum_{t=1}^T c_t(\mu_t) - c_t(\mu_{t+1}) \\ &\leq \sum_{t=1}^T \nabla c_t(\mu_t) \cdot (\mu_t - \mu_{t+1}) \quad (\because c_t \text{ is convex}) \\ &\leq \sum_{t=1}^T \|\nabla c_t(\mu_t)\| \|\mu_t - \mu_{t+1}\| \\ &\leq \sum_{t=1}^T \|2(f_t - \mu_t)\| \cdot \|\mu_t - \mu_{t+1}\|. \end{aligned}$$

Now, we have

$$\begin{aligned} \|\mu_t - \mu_{t+1}\| &= \left\| \mu_t - \frac{t\mu_t + f_t}{t+1} \right\| \\ &\leq \frac{1}{t+1} (\|\mu_t\| + \|f_t\|) \\ &\leq \frac{2}{t}. \end{aligned}$$

Thus, the regret is bounded by  $4\rho(T)$ . ■

**Proof:** (**Lemma 8**) We may assume without loss of generality that  $\mu = 0$ : using the vectors  $f_t - \mu$  instead of  $f_t$  doesn't change the value of  $\rho(T)$ . We have

$$\begin{aligned}\rho(T) &= \sum_{t=1}^T \frac{1}{t} \|f_t - F_t/t\| \\ &\leq \sum_{t=1}^T \frac{1}{t} \left[ \|f_t\| + \frac{1}{t} \|F_t\| \right] \\ &\leq \sum_{t=1}^T \left[ \frac{1}{t} \|f_t\| + \frac{1}{t^2} \sum_{\tau=1}^{t-1} \|f_\tau\| \right] \\ &\leq \sum_{t=1}^T \frac{2}{t} \|f_t\| \quad \left( \because \sum_{\tau=t+1}^T \frac{1}{t^2} \leq \frac{1}{t} \right) \\ &\leq \sqrt{\left[ \sum_{t=1}^T \|f_t\|^2 \right] \left[ \sum_{t=1}^T \frac{4}{t^2} \right]} \quad (\text{Cauchy-Schwarz}) \\ &\leq 3 \sqrt{\sum_{t=1}^T \|f_t\|^2},\end{aligned}$$

as required. ■

The projection lemma which follows is a well-known fact from convex optimization theory. We include the proof for completeness.

**Lemma 9 (Projection lemma)** *Let  $K$  be a convex set, and let  $x$  and  $y$  be any two points. Let  $x'$  and  $y'$  be their respective projections on  $K$ . Then*

$$\|x' - y'\| \leq \|x - y\|.$$

**Proof:** Assume that  $x' \neq y'$ , otherwise the inequality is trivial. By the properties of projections on convex sets, we have

$$(x - x') \cdot (y' - x') \leq 0 \text{ and } (y - y') \cdot (x' - y') \leq 0. \quad (5)$$

Consider the line  $\ell$  passing through  $x'$  and  $y'$ , and consider the projections  $x''$  and  $y''$  of  $x$  and  $y$  respectively on this line. The inequalities (5) imply that along  $\ell$ , the order of the points is  $(x'', x', y', y'')$ . Thus, we have

$$\|x' - y'\| \leq \|x'' - y''\| \leq \|x - y\|,$$

where the last inequality follows because the projection of any line segment on any line is no longer than the segment itself. ■

## 5 Analysis of the Variation MW algorithm

The analysis of the Variation MW is straightforward, though complicated somewhat due to heavy algebraic manipulations. We outline the main ideas in the analysis now. Our starting point is Lemma 10, a well-known bound which relates the regret of the Multiplicative Weights algorithm with the expected squared losses of the experts (the expectation being taken under the distributions generated by the algorithm).

Next, we make crucial use of the fact that the Multiplicative Weights algorithm puts exponentially higher weight on experts with lower cost than those with higher costs. Since we explicitly factor in the variation in the costs of each expert before computing their exponential weights, eventually the algorithm starts to concentrate all the weight on experts with lower cost and lower variation. This yields the desired regret bound.

We now describe a regret bound on the performance of the Multiplicative Weights algorithm. This bound is well-known (see, for e.g. [4, 2]), we include the short proof for completeness.

**Lemma 10** *Suppose in round  $t$  of the expert prediction problem, expert  $i$  incurs cost  $g_t(i)$ , where  $|g_t(i)| \leq M$ . Consider the Multiplicative Weights algorithm, that in round  $t$  chooses expert  $i$  with probability  $x_t(i) \propto \exp(-\eta \sum_{\tau=1}^{t-1} g_\tau(i))$ . Then, if  $\eta \leq 1/M$ ,*

$$\text{Regret} \leq \eta \sum_{t=1}^T g_t^2 \cdot x_t + \frac{\log n}{\eta}.$$

**Proof:** Let  $w_t(i) = \exp(-\eta \sum_{\tau=1}^{t-1} g_\tau(i))$ , and let  $Z_t = \sum_i w_t(i)$ . Then the distribution on the experts at time  $t$  is exactly  $w_t/Z_t$ . We think of  $Z_t$  as a potential function, and track how it changes over time. Initially,  $Z_1 = n$ . We have

$$\begin{aligned}Z_{t+1} &= \sum_i w_t(i) \exp(-\eta g_t(i)) \\ &\leq \sum_i w_t(i) (1 - \eta g_t(i) + \eta^2 g_t(i)^2) \quad (6) \\ &= Z_t (1 - \eta(g_t \cdot x_t) + \eta^2(g_t^2 \cdot x_t)) \\ &\leq Z_t \exp(-\eta(g_t \cdot x_t) + \eta^2(g_t^2 \cdot x_t)).\end{aligned}$$

In (6), we used the fact that for  $|x| \leq 1$ , we have  $\exp(x) \leq 1 + x + x^2$ . Thus, by induction, we have

$$Z_{T+1} \leq n \exp\left(-\eta \sum_{t=1}^T (g_t \cdot x_t) + \eta^2 \sum_{t=1}^T (g_t^2 \cdot x_t)\right).$$

Also, for any expert  $i$  we have the bound

$$Z_{T+1} \geq w_{T+1}(i) = \exp\left(-\eta \sum_{\tau=1}^T g_\tau(i)\right).$$

Putting these two inequalities together, taking logarithms and simplifying, we get the desired bound on the regret. ■

For our analysis, we use a slightly different notion of variation of the experts' costs: for any round  $t$  and any expert  $i$ , define

$$Q_t(i) = \sum_{\tau=1}^{t-1} (f_\tau(i) - \mu_\tau(i))^2.$$

Recall that the usual definition of variation of an experts cost up to the  $t^{\text{th}}$  round is simply

$$\text{VAR}_t(i) = \sum_{\tau=1}^t (f_\tau(i) - \mu_t^\star(i))^2,$$

where  $\mu_t^*(i) = \frac{1}{t} \sum_{\tau=1}^t f_t(i)$ . But it is easily seen from (the 1 dimensional version of) Lemmas 7 and 8 that

$$Q_t(i) \leq \text{VAR}_t(i) + 12\sqrt{\text{VAR}_t(i)}. \quad (7)$$

and thus  $Q_t(i)$  can serve as a proxy for the true variation (up to constant factors).

Recall that  $\ell_t$  is the best expert till time  $t$ , and  $\text{VAR}_T^{\max} = \max_{t \leq T} \{\text{VAR}_t(\ell_t)\}$ . Define  $Q_T^{\max} = \max_{t \leq T} Q_t(\ell_t)$ . Then, we have that

$$Q_T^{\max} \leq 4\text{VAR}_T^{\max},$$

assuming that  $\text{VAR}_T^{\max} \geq 16$ . Then, the following Lemma combined with inequality (7) implies Theorem 4.

**Lemma 11** *Let  $f_t$ , for  $t = 1, 2, \dots, T$ , be a sequence of cost vectors to the experts so that  $f_t(i) \in [0, 1]$ . Let  $\ell_t$  be the best expert at time  $t$ , and let  $Q$  be an upper bound on  $Q_T^{\max} = \max_t \{Q_t(\ell_t)\}$ . Then setting  $\eta = \min\{\sqrt{\log(n)/4Q}, 1/10\}$ , the regret of the Variation MW algorithm is bounded by*

$$\text{Regret} \leq 4\sqrt{Q \log(n)} + 10 \log(n).$$

**Proof:** Define  $g_t = \tilde{f}_t - \alpha_t \vec{1}$ , where  $\alpha_t = \mu_t(\ell_t) + \frac{4\eta}{t} Q_t(\ell_t)$ , and  $\vec{1}$  is the all 1's vector. Note that for any  $i$ ,

$$\exp\left(-\eta \sum_{\tau=1}^{t-1} g_\tau(i)\right) = \frac{1}{Z} \exp\left(-\eta \sum_{\tau=1}^{t-1} \tilde{f}_\tau(i)\right),$$

where  $Z$  is a scaling constant independent of  $i$ . Hence, scaling either the weights  $\exp(-\eta \sum_{\tau=1}^{t-1} g_\tau(i))$  or the weights  $\exp(-\eta \sum_{\tau=1}^{t-1} \tilde{f}_\tau(i))$  to sum up to 1 yields the same distribution, viz.  $x_t$ .

Since we assumed that the  $f_t(i) \in [0, 1]$ , we conclude that  $g_t(i) \in [-2, 2]$  (since  $4\eta \leq 1$ ). Applying Lemma 10 to the sequence of cost vectors  $g_t$ , we get the following regret bound, where  $\ell_T$  is the final best expert:

$$\sum_{t=1}^T \tilde{f}_t \cdot x_t - \sum_{t=1}^T \tilde{f}_t(\ell_T) \leq \eta \sum_{t=1}^T g_t^2 \cdot x_t + \frac{\log n}{\eta}.$$

Here, we used the fact that the  $\sum_{t=1}^T \alpha_t \vec{1} \cdot x_t = \sum_{t=1}^T \alpha_t$ . Simplifying using the definition of  $\tilde{f}_t$ , we get

$$\begin{aligned} & \sum_{t=1}^T f_t \cdot x_t - \sum_{t=1}^T f_t(\ell_T) \\ & \leq \eta \sum_{t=1}^T g_t^2 \cdot x_t + \frac{\log n}{\eta} \\ & \quad - 4\eta \sum_{t=1}^T (f_t - \mu_t)^2 \cdot x_t + 4\eta \sum_{t=1}^T (f_t(\ell_T) - \mu_t(\ell_T))^2 \\ & \leq \eta \sum_{t=1}^T [g_t^2 - 4(f_t - \mu_t)^2] \cdot x_t + 4\eta(Q+1) + \frac{\log n}{\eta}, \end{aligned} \quad (8)$$

since  $\sum_{t=1}^T (f_t(\ell_T) - \mu_t(\ell_T))^2 \leq Q_T(\ell_T) + 1 \leq Q + 1$ .

The following lemma bounds the first term in (8). The proof is a straightforward calculation, and so we defer its proof to after the present proof.

**Lemma 12** *If  $\eta \leq 1/10$ , then for any  $i$ , we have*

$$g_t^2(i) - 4(f_t(i) - \mu_t(i))^2 \leq 2(\mu_t(i) - \alpha_t)^2.$$

Plugging this bound into (8), we get that

$$\text{Regret} \leq 2\eta \sum_{t=1}^T (\mu_t - \alpha_t \vec{1})^2 \cdot x_t + \frac{\log n}{\eta} + 4\eta(Q+1). \quad (9)$$

We now proceed to bound  $\sum_{t=1}^T (\mu_t - \alpha_t \vec{1})^2 \cdot x_t$ . We bound each term in the summation separately. For any  $t \leq \frac{\log n}{\eta}$ , we simply bound  $|\mu_t(i) - \alpha_t| \leq 2$  and hence we have  $(\mu_t - \alpha_t \vec{1})^2 \cdot x_t \leq 2$ .

Now let  $t > \frac{\log n}{\eta}$ . For convenience of notation, we drop the subscript  $t$  from  $x_t(i)$  and refer to them as  $x(i)$ .

$$\begin{aligned} & (\mu_t - \alpha_t \vec{1})^2 \cdot x \\ &= \sum_{i: \mu_t(i) \leq \alpha_t} (\mu_t(i) - \alpha_t)^2 x(i) + \sum_{i: \mu_t(i) > \alpha_t} (\mu_t(i) - \alpha_t)^2 x(i) \\ &\leq \sum_{i: \mu_t(i) \leq \alpha_t} \left[ \frac{4\eta}{t} Q_t(\ell_t) \right]^2 x(i) + \sum_{i: \mu_t(i) > \alpha_t} (\mu_t(i) - \alpha_t)^2 x(i) \end{aligned} \quad (10)$$

$$\leq \left[ \frac{4\eta}{t} Q_t(\ell_t) \right]^2 + \sum_{i: \mu_t(i) > \alpha_t} (\mu_t(i) - \alpha_t)^2 x(i) \quad (11)$$

Here, (10) follows because when  $\mu_t(i) \leq \alpha_t = \mu_t(\ell_t) + \frac{4\eta}{t} Q_t(\ell_t)$ , we have  $|\mu_t - \alpha_t| \leq \frac{4\eta}{t} Q_t(\ell_t)$  since  $\mu_t(i) \geq \mu_t(\ell_t)$ .

We now bound each term of (11) separately. The proof of the following lemma is a straightforward calculation and we defer it to after the present proof.

**Lemma 13** *The first term of (11), summed over all  $t$ , can be bounded as:*

$$\sum_{t=1}^T \left[ \frac{4\eta}{t} Q_t(\ell_t) \right]^2 \leq 32\eta^2 Q.$$

The hard part is to bound the second term of (11). We now proceed to do so. The intuition in the following analysis is that the Variation MW algorithm tends to concentrate exponentially high weight on the experts that have low cost.

Let  $I$  be the index set of all  $i$  such that  $\mu_t(i) > \alpha_t$ . Note that  $\ell_t \notin I$ . Now, we have  $x(i) \propto \exp(-\eta t \mu_t(i) - 4\eta^2 Q_t(i))$ , and thus  $x(\ell_t) \propto \exp(-\eta t \alpha_t)$ . Thus,  $x(i)$  can be written as:

$$\begin{aligned} x(i) &= \frac{\exp(-\eta t \mu_t(i) - 4\eta^2 Q_t(i))}{\exp(-\eta t \alpha_t) + \sum_{j \neq \ell_t} \exp(-\eta t \mu_t(j) - 4\eta^2 Q_t(j))} \\ &= \frac{\lambda(i) \exp(-\eta t (\mu_t(i) - \alpha_t))}{1 + \sum_{j \neq \ell_t} \lambda(j) \exp(-\eta t (\mu_t(j) - \alpha_t))}, \end{aligned}$$

where  $\lambda(i) = \exp(-4\eta^2 Q_t(i))$ . Note that all  $\lambda(i) \in (0, 1]$ . Define, for all  $i$ ,  $d(i) = (\mu_t(i) - \alpha_t)$ . Note that for  $i \in I$ ,  $d(i) \in [0, 1]$ . Thus, we have

$$\sum_{i \in I} d(i)^2 x(i) = \sum_{i \in I} \frac{\lambda(i) d(i)^2 \exp(-\eta t d(i))}{1 + \sum_{j \neq \ell_t} \lambda(j) \exp(-\eta t d(j))}.$$

To upper bound  $\sum_{i \in I} d(i)^2 x(i)$ , we can neglect the factors in the denominator which depend on  $i \notin I \cup \{\ell_t\}$ ; this only increases the value. Let  $d^I$  and  $\lambda^I$  be the vectors  $d$  and  $\lambda$  restricted to the index set  $I$ . Define the function  $h : (0, 1]^{|I|} \times [0, 1]^{|I|} \rightarrow \mathbb{R}$  as

$$h(\lambda^I, d^I) = \sum_{i \in I} \frac{\lambda(i)d(i)^2 \exp(-\eta t d(i))}{1 + \sum_{j \in I} \lambda(j) \exp(-\eta t d(j))}.$$

This maximum value of this function on its domain gives an upper bound on the expression above.

**Lemma 14** For  $t > \frac{\log n}{\eta}$ , and for any  $(\lambda^I, d^I) \in (0, 1]^{|I|} \times [0, 1]^{|I|}$ , we have

$$h(\lambda^I, d^I) \leq \frac{2 \log^2 n}{\eta^2 t^2}.$$

Putting Lemmas 13 and 14 together, we have that

$$\begin{aligned} \sum_{t=1}^T (\mu_t - \alpha_t \vec{1})^2 \cdot x_t &\leq \sum_{t \leq \frac{\log n}{\eta}} 2 + \sum_{t=1}^T \left[ \frac{4\eta}{t} Q_t(\ell_t) \right]^2 \\ &\quad + \sum_{t > \frac{\log n}{\eta}} \frac{2 \log^2 n}{\eta^2 t^2} \\ &\leq 32\eta^2 Q + \frac{4 \log n}{\eta}. \end{aligned}$$

Plugging this bound into (9), we get

$$\text{Regret} \leq \frac{\log n}{\eta} + 64\eta^3 Q + 8 \log(n) + 4\eta(Q+1).$$

Now, if we set  $\eta = \{\sqrt{\log n / 4Q}, 1/10\}$ , we get that the regret is bounded by

$$\text{Regret} \leq 4\sqrt{Q \cdot \log n} + 10 \log(n). \quad \blacksquare$$

Again, it may not be possible to get an upper bound on  $Q_T^{\max}$  *a priori*, but we can use the same  $\eta$ -halving idea (start with  $\eta = 1/10$ , and halve  $\eta$  as soon as this maximum quadruples, and restart the algorithm) and get regret that bounded by

$$\text{Regret} \leq O\left(\sqrt{Q_T^{\max} \log(n)} + \log(Q_T^{\max}) \log(n)\right).$$

The details of this bound are standard and are hence omitted from this extended abstract.

We now give the omitted proofs of Lemmas 12, 13, and 14.

**Proof: (Lemma 12)**

We have:

$$\begin{aligned} g_t(i)^2 &= (f_t(i) - \alpha_t + 4\eta(f_t(i) - \mu_t(i))^2)^2 \\ &= (f_t(i) - \alpha_t)^2 + 8\eta(f_t(i) - \alpha_t)(f_t(i) - \mu_t(i))^2 \\ &\quad + 16\eta^2(f_t(i) - \mu_t(i))^4 \\ &\leq (f_t(i) - \alpha_t)^2 + (16\eta + 16\eta^2)(f_t(i) - \mu_t(i))^2 \end{aligned} \tag{12}$$

$$\leq 2(\mu_t(i) - \alpha_t)^2 + (2 + 16\eta + 16\eta^2)(f_t(i) - \mu_t(i))^2 \tag{13}$$

$$\leq 2(\mu_t(i) - \alpha_t)^2 + 4(f_t(i) - \mu_t(i))^2. \tag{14}$$

Here, inequality (12) follows because  $|f_t(i) - \mu_t(j)| \leq 1$  for any  $i, j$ , and  $|f_t(i) - \alpha_t| \leq 2$ , inequality (13) follows from the fact that  $(a+b)^2 \leq 2a^2 + 2b^2$  for any real numbers  $a, b$ , and inequality (14) follows since  $16\eta + 16\eta^2 \leq 2$  if  $\eta \leq 1/10$ . The lemma follows.  $\blacksquare$

**Proof: (Lemma 13)**

Note that for  $t \leq Q$ ,  $Q_t(\ell_t) = \sum_{\tau=1}^{t-1} (f_\tau(i) - \mu_\tau(i))^2 \leq t$ , and for  $t > Q$ ,  $Q_t(\ell_t) \leq Q$ . Thus we have

$$\sum_{t=1}^T \left[ \frac{4\eta}{t} Q_t(\ell_t) \right]^2 \leq 16\eta^2 \cdot \left[ \sum_{t \leq Q} 1^2 + \sum_{t > Q} \frac{Q^2}{t^2} \right] \leq 32\eta^2 Q. \quad \blacksquare$$

**Proof: Lemma 14)**

Let  $S = \{i : d(i) \leq \frac{\log n}{\eta t}\}$ , and let  $S' = I \setminus S$ . We upper bound  $h(\lambda^I, d^I)$  as follows:

$$\begin{aligned} h(\lambda^I, d^I) &\leq \sum_{i \in S} \frac{\lambda(i)d(i)^2 \exp(-\eta t d(i))}{\sum_{j \in S} \lambda(j) \exp(-\eta t d(j))} \\ &\quad + \sum_{i \in S'} \lambda(i)d(i)^2 \exp(-\eta t d(i)) \\ &\leq \max_{i \in S'} \left\{ \frac{\lambda(i)d(i)^2 \exp(-\eta t d(i))}{\lambda(i) \exp(-\eta t d(i))} \right\} \end{aligned} \tag{15}$$

$$\begin{aligned} &\quad + \sum_{i \in S'} \frac{\log^2 n}{\eta^2 t^2} \exp(-\log n) \\ &\leq \frac{2 \log^2 n}{\eta^2 t^2}. \end{aligned} \tag{16}$$

In (15) we use the inequality  $\frac{\sum_{i=1}^n a_i}{\sum_{i=1}^n b_i} \leq \max_{i \leq n} \frac{a_i}{b_i}$  for positive reals  $a_i$  and  $b_i$ . In (16), we used the following facts (a)  $\lambda(i) \leq 1$ , and (b) the function  $x^2 \exp(-\eta t x)$  has a negative derivative (and is thus decreasing) when  $x > \frac{2}{\eta t}$ , and thus its maximum over the range  $[\frac{\log n}{\eta t}, 1]$  is obtained at  $\frac{\log n}{\eta t}$ .  $\blacksquare$

## 6 Conclusions and Future Work

In this paper, we investigated the possibility of bounding the regret of online learning algorithms by terms which depend on the variation of the cost sequence, rather than the number of prediction rounds. We analyzed two algorithms, Lazy Projection and Variation MW, and showed that these algorithms obtain variation-bounded regret. Such bounds are significant not only because they show that it is possible to suffer much less regret than previously believed when the cost sequence is particularly benign, but also because they match the regret bounds of natural regret minimizing algorithms in the stochastic setting of independent cost functions from a fixed distribution.

We believe that this work opens up many new directions for future research, all related to bounding the regret in terms of the variation of the cost sequence in the various different scenarios in which regret minimizing algorithms have been devised: bandit settings, strictly convex cost functions, online convex optimization and so on. We conjecture in all such scenarios, it is possible to get variation-bounded regret.

Specifically, we conjecture that any dependence on  $T$ , the number of prediction rounds, in the regret bound can be replaced by the same dependence on the variation of the cost sequence. In other scenarios, the variation needs to be defined carefully in settings in which it is not natural or obvious, such as in the case of online convex optimization.

## Acknowledgements

We thank Martin Zinkevich for initial discussions on the possibility of variation bounds on the regret.

## References

- [1] Chamy Allenberg-Neeman and Benny Neeman. Full information game with gains and losses. In *15'th International Conference on Algorithmic Learning Theory*, 2004.
- [2] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2003.
- [3] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.
- [4] Nicolò Cesa-Bianchi, Yishay Mansour, and Gilles Stoltz. Improved second-order bounds for prediction with expert advice. *Mach. Learn.*, 66(2-3):321–352, 2007.
- [5] T. Cover. Universal portfolios. *Math. Finance*, 1:1–19, 1991.
- [6] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, 1997.
- [7] James Hannan. Approximation to bayes risk in repeated play. In M. Dresher, A. W. Tucker, and P. Wolfe, editors, *Contributions to the Theory of Games, volume III*, pages 97–139, 1957.
- [8] D. P. Helmbold, J. Kivinen, and M. K. Warmuth. Relative loss bounds for single neurons. *IEEE Transactions on Neural Networks*, 10(6):1291–1304, November 1999.
- [9] Mark Herbster and Manfred K. Warmuth. Tracking the best linear predictor. *Journal of Machine Learning Research*, 1:281–309, 2001.
- [10] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.
- [11] Jyrki Kivinen and Manfred K. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Inf. Comput.*, 132(1):1–63, 1997.
- [12] Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.
- [13] V. Vovk. A game of prediction with expert advice. *J. Comput. Syst. Sci.*, 56(2):153–173, 1998.
- [14] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, pages 928–936, 2003.