Stochastic and Adversarial Online Learning without Hyperparameters

Authored by:

Kwabena A. Boahen
Ashok Cutkosky

Abstract

Most online optimization algorithms focus on one of two things: performing well in adversarial settings by adapting to unknown data parameters (such as Lipschitz constants), typically achieving $O(sqrt{T})$ regret, or performing well in stochastic settings where they can leverage some structure in the losses (such as strong convexity), typically achieving $O(log(T))$ regret. Algorithms that focus on the former problem hitherto achieved $O(sqrt{T})$ in the stochastic setting rather than $O(log(T))$. Here we introduce an online optimization algorithm that achieves $O(log4(T))$ regret in a wide class of stochastic settings while gracefully degrading to the optimal $O(sqrt{T})$ regret in adversarial settings (up to logarithmic factors). Our algorithm does not require any prior knowledge about the data or tuning of parameters to achieve superior performance.

1 Paper Body

Most online optimization algorithms focus on one of two things: performing well in adversarial settings by adapting to ?unknown data parameters (such as Lipschitz constants), typically achieving O(T) regret, or performing well in stochastic settings where they can leverage some structure in the losses (such as strong convexity), typically achieving O(log(T ?)) regret. Algorithms that focus on the former problem hitherto achieved O(T) in the stochastic setting rather than O(log(T)). Here we introduce an online optimization algorithm that achieves O(log4 (T)) regret? in a wide class of stochastic settings while gracefully degrading to the optimal O(T) regret in adversarial settings (up to logarithmic factors). Our algorithm does not require any prior knowledge about the data or tuning of parameters to achieve superior performance.

1

Extending Adversarial Algorithms to Stochastic Settings

The online convex optimization (OCO) paradigm [1, 2] can be used to model a large number of scenarios of interest, such as streaming problems, adversarial environments, or stochastic optimization. In brief, an OCO algorithm plays T rounds of a game in which on each round the algorithm outputs a vector wt in some convex space W , and then receives a loss function ‘t : W ? R that is convex. The algorithm?s objective is to minimize regret, which is the total loss of all rounds relative to w? , PT the minimizer of t=1 ‘t in W : RT (w?) =

T X

‘t (wt) ? ‘t (w?)

t=1

OCO algorithms typically either make as few as possible assumptions about the ‘t while attempting to perform well (adversarial settings), or assume that the ‘t have some particular structure that can be leveraged to perform much ? better (stochastic settings). For the adversarial setting, the minimax optimal regret is O(BLmax T), where B is the diameter of W and Lmax is the maximum Lipschitz constant of the losses [3]. A wide variety of algorithms achieve this bound without prior knowledge of one or both of B and Lmax [4, 5, 6, 7], resulting in hyperparameter-free algorithms. In the stochastic setting, it was recently shown that for a class of problems (those satisfying the so-called Bernstein condition), one can achieve regret O(dBLmax log(T)) where W ? Rd using the M ETAG RAD algorithm [8, 9]. This approach requires knowledge of the parameter Lmax . In this paper, we extend an algorithm for the parameter-free adversarial setting [7] to the stochastic setting, achieving both optimal regret in adversarial settings as well as logarithmic regret in a wide class of stochastic settings, without needing to tune parameters. Our class of stochastic settings is those for which E[¿t (wt)] is aligned with wt ? w? , quantified by a value ? that increases with 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

increasing alignment. We call losses in this class ?-acutely convex, and show that a single quadratic lower bound on the average loss is sufficient to ensure high ?. This paper is organized as follows. In Section 2, we provide an overview of our approach. In Section 3, we give explicit pseudo-code and prove our regret bounds for the adversarial setting. In Section 4, we formally define ?-acute convexity and prove regret bounds for the acutely convex stochastic setting. Finally, in Section 5, we give some motivating examples of acutely convex stochastic losses. Section 6 concludes the paper.

2

Overview of Approach

Before giving the overview, we fix some notation. We assume our domain W is a closed convex subset of a Hilbert space with 0 ? W . We write gt to be an arbitrary subgradient of ‘t at wt for all t, which we denote by gt ? ¿t (wt). Lmax is the maximum Lipschitz constant?of all the ‘t , and B is the diameter of the space W . The norm k ? k we use is the 2-norm: kwk = w ? w. We observe PT that since each ‘t is convex, we have RT (w?) ? t=1 gt (wt ? w?). We will make heavy use of this PT inequality; every regret bound we state will in fact be an upper bound on t=1 gt (wt ? w?). Finally, Pt ? to suppress logarithmic terms in we use a compressed sum notation g1:t = t0 =1 gt0 , and we use O big-Oh notation. All proofs omitted from the main text appear in the appendix. Our algorithm works by trading off some performance in order to avoid knowledge of problem parameters. Prior analysis of the M ETAG RAD q algorithm [9] showed that any algorithm guaranteeing P T ? 2 ? RT (w?) = O will obtain logarithmic regret for stochastic settings t=1 (gt ? (wt ? w)) satisfying the Bernstein condition. We will instead guarantee the weaker regret bound: ?v ? u T u X ? ?tLmax RT (w?) ? O kgt kkwt ? w? k2 ?

(1)

t=1

? which we will show in turn implies T regret in adversarial settings and logarithmic regret for acutely convex stochastic settings. Although (1) is weaker than the M ETAG RAD regret bound, we can obtain it without prior knoweldge. In order to come up with an algorithm that achieves the bound (1), we interpret it as the square root of E[kw ? w? k2], where w takes on value wt with probability proportional to kgt k. This allows us to use the bias-variance decomposition to write (1) as: v ? u T uX p ? ? ? ?kw ? wk Lmax kgk1:T + t RT (w) ? O Lmax kgt kkwt ? wk2 ? ?

(2)

t=1 PT

kg kw

t t t=1 where w = . Certain algorithms for unconstrained OCO can achieve RT (u) = p kgk1:T ? O(kukLmax kgk1:T) simultaneously for all u ? W [10, 6, 11, 7]. Thus if we knew w ahead of time, we could p translate the predictions of one such algorithm by w to abtain RT (w?) ? ? ? O(kw ? wkLmax kgk1:T), the bias term of (2). We do not know w, but we can estimate it over time. Errors in the estimation procedure will cause us to incur the variance term of (2). We implement this strategy by modifying F REE R EX [7], an unconstrained OCO algorithm that does not require prior knowledge of any parameters.

Our modification to F REE R EX is very simple: we set wt = w ?t + wt?1 where w ?t is the tth output of F REE R EX, and wt?1 is (approximately) a weighted average of the previous vectors w1 , . . . , wt?1 with the weight of wt equal to kgt k. This wt offset can be viewed as a kind of momentum term that accelerates us towards optimal points when the losses are stochastic (which tends to cause correlated wt and therefore large offsets), but has very little effect when the losses are adversarial (which tends to cause uncorrelated wt and therefore small offsets). 2

3

F REE R EX M OMENTUM

In this section, we explicitly describe and analyze our algorithm, F REE R EX M OMENTUM, a modification of F REE R EX. F REE R EX is a Follow-the-Regularized-Leader (FTRL) algorithm, which means that for all t, there is some regularizer function ?t such that wt+1 = argminW ?t (w) + g1:t ? w. ? 5 Specifically, F REE R EX uses ?t = at ?t ?(at w), where ?(w) = (kwk + 1) log(kwk + 1) ? kwk and ?t and at are specific numbers that grow over time as specified in Algorithm 1. F REE R EX M O MENTUM ?s predictions are given by offsetting F REE R EX ?s predictions wt+1 by a momentum term P t?1 t0 =1 kgt0 kwt . 1+kgk1:t

wt =

We accomplish this by shifting the regularizers ?t by wt , so that F REE R EX M O MENTUM is FTRL with regularizers ?t (w ? w t).

Algorithm 1 F REE R EX M OMENTUM Initialize: ?12 ? 0, a0 ? 0, w1 ? 0, L0 ? 0, ?(w) = (kwk + 1) log(kwk + 1) ? kwk 0 for t = 1 to T do Play wt Receive subgradient gt ? ¿t (wt) Lt ? max(L t?1 , kgt k). // Lt = maxt0 ?t kgt k 1 ?t2

? max

1 2 ?t?1

+ 2kgt k2 , Lt kg1:t k .

2 at ? max(a t?1 , 1/(Lt ?t)) P

wt ?

t?1 t0 =1

kgt0 kwt 1+kgk1:t h

wt+1 ? argminW end for 3.1

?

5?(at (w?wt) at ?t

+ g1:t ? w

i

Regret Analysis

We leverage the description of F REE R EX M OMENTUM in terms of shifted regularizers to prove a regret bound of the same form as (1) in four steps: 1. From [7] Theorem 13, we bound the regret by RT (w?) ?

T X

gt ? (wt ? w?)

t=1

? ?T (w?) +

T X

+ + + ?t?1 (wt+1) ? ?t+ (wt+1) + gt ? (wt ? wt+1)

t=1

+ ?T+ (w?) ? ?T (w?) +

T ?1 X

+ + ?t+ (wt+2) ? ?t (wt+2)

t=1

?

t?1) where ? 5?(ata(w?w is a version of ?t shifted by wt?1 instead of wt , and t ?t + + wt+1 = argminW ?t (w) + g1:t w. This breaks the regret out into two sums, one in which + + we have the term ?t?1 (wt+1) ? ?t+ (wt+1) for which the two different functions are shifted + + by the same amount, and one with the term ?t+ (wt+2) ? ?t (wt+2), for which the functions

?t+ (w)

are shifted differently, but the arguments are the same. 2. Because ?t?1 and ?t+ are shifted by the same amount, the regret analysis for F REE R EX in [7]papplies to the second line of the regret bound, yielding a quantity similar to kw? ? wT k Lmax kgk1:T . 3. Next, we analyze the third line. We show that wt ? wt?1 q cannot be too big, and use this PT 2 observation to bound the third line with a quantity similar to t=1 Lmax kgt k(wt ? w T) . At this point we have enough results to prove a bound of the form (2) (see Theorem 1). 4. Finally, we perform some algebraic manipulation on the bound from the first three steps to obtain a bound of the form (1) (see Corollary 2). 3

The details of Steps 1-3 procedure are in the appendix, resulting in Theorem 1, stated below. Step 4 is carried out in Corollary 2, which follows. 1:T Theorem 1. Let ?(w) = (kwk+1) log(kwk+1)?kwk. Set Lt = maxt0 ?t kgt0 k, and QT = 2 kgk Lmax . Define ?1t and at as in the pseudo-code for F REE R EX M OMENTUM (Algorithm 1). Then the regret of F REE R EX M OMENTUM is bounded by: ? ? T X 5 Lmax 2Lmax gt ?(wt ?w?) ? B log(BaT +1) ?(QT (w? ?wT))+405Lmax +2Lmax B+3 ? QT ?T 1 + L1 t=1

v u u +t2Lmax

kwT

k2

+

T X

! kgt kkwt ? wT

k2

2 + log

t=1

1 + kgk1:T 1 + kg1 k

log(BaT + 1)

Corollary 2. Under the assumptions and notation of Theorem 1, the regret of F REE R EX M OMENTUM is bounded by: v ! u T T u X X ? ? kgt kkw? ? wt k2 log(2BT + 1)(2 + log(T)) gt ? (wt ? w) ? 2 5tLmax kw? k2 + t=1

t=1

? Lmax 2Lmax B log(2BT + 1) + 405Lmax + 2Lmax B + 3 ? 1 + L1

Observe that since wt and w? are both in W , kw? k and kwt ? w? k?both are at most B, so that ? Corollary 2 implies that F REE R EX M OMENTUM achieves O(BL max T) regret in the worst-case, which is optimal up to logarithmic factors. 3.2

Efficient Implementation for L? Balls

A carefulhreader may notice that the i procedure for F REE R EX M OMENTUM involves computing ? 5?(at (w?wt) argminW + g1:t ? w , which may not be easy if the solution wt+1 is on the boundary at ?t of W . When the wt+1 is not on the boundary of W , then we have a closed-form update:

g1:t ?t kg1:t k ? wt+1 = wt ? ?1 (3) exp at kg1:t k 5 However, when wt+1 lies on the boundary of W , it is not clear how to compute it for general W . In Qd this section we offer a simple strategy for the case that W is an L? ball, W = i=1 [?b, b]. In this setting, we can use the standard trick (e.g. see [12]) of running a separate copy of F REE R EX M OMENTUM for each coordinate. That is, we observe that RT (w?) ?

T X

gt ? (wt ? u) =

t=1

d X T X
gt,i (wt,i ? ui)
(4)
i=1 t=1
so that if we run an independent online learning algorithm on each coordinate, using the coordinates of the gradients gt,i as losses, then the total regret is at most the sum of the individual regrets. More detailed pseudocode is given in Algorithm 2. Coordinate-wise F REE R EX M OMENTUM is easily implementable in time O(d) per update because the F REE R EX M OMENTUM update is easy to perform in one dimension: if the update (3) is outside the domain [?b, b], simply set wt+1 to b or ?b, whichever is closer to the unconstrained update. Therefore, coordinate-wise F REE R EX M OMENTUM can be computed in O(d) time per update. We bound the regret of coordinate-wise F REE R EX M OMENTUM using Corollary 2 and Equation (4), resulting the following Corollary. 4

Algorithm 2 Coordinate-Wise F REE R EX M OMENTUM Initialize: w1 = 0, d copies of F REE R EX M OMENTUM, F1 ,. . . ,Fd , where each Fi uses domain W = [?b, b]. for t = 1 to T do Play wt , receive subgradient gt . for i = 1 to d do Give gt,i to Fi . Get wt+1,i ? [?b, b] from Fi . end for end for Corollary 3. The regret of coordinate-wise F REE R EX M OMENTUM is bounded by: v ! u T T u X X ? ? t ? 2 ? 2 g ? (w ? w) ? 2 5 dL dkw k + kg kkw ? w k log(2T b + 1)(2 + log(T)) t

max

t

t

t=1

t=1

+ 405dLmax + 2Lmax db + 3d

4

t

? Lmax 2Lmax ? b log(2bT + 1) 1 + L1

Logarithmic Regret in Stochastic Problems

In this section we formally define ?-acute convexity and show that F REE R EX M OMENTUM achieves logarithmic regret for ?-acutely convex losses. As a warm-up, we first consider the simplest case in which the loss functions ‘t are fixed, ‘t = ‘ for all t. After showing logarithmic regret for this case, we will then generalize to more complicated stochastic settings. Intuitively, an acutely convex loss function ‘ is one for which the gradient gt is aligned with the vector wt ? w? where w? = argmin ‘, as defined below. Definition 4. A convex function ‘ is ?-acutely convex on a set W if ‘ has a global minimum at some w? ? W and for all w ? W , for all subgradients g ? ¿(w), we have g ? (w ? w?) ? ?kgkkw ? w? k2 With this definition in hand, we can show logarithmic regret in the case where ‘t = ‘ for all t for some ?-acutely convex function ‘. From Corollary 2, with w? = argmin ‘, we have ?v !? u T T u X X ? ?tLmax kw? k2 + gt ? (wt ? w?) ? O kgt kkw? ? wt k2 ? t=1

t=1

?v u u ? ? ? O tLmax

!? T X 1 kw? k + gt ? (w? ? wt) ? ? t=1

(5)

? notation suppresses terms whose dependence on T is at most O(log2 (T)). Now we Where the O need a small Proposition: Proposition 5. If a, b, c and d are non-negative constants such that ? x ? a bx + c + d Then

? x ? 4a2 b + 2a c + 2d

PT Applying Proposition 5 to Equation (5) with x = t=1 gt ? (wt ? w?) yields

? ? Lmax kw k RT (u) ? O ? 5

? again suppresses logarithmic terms, now with dependence on T at most O(log4 (T)). where the O Having shown that F REE R EX M OMENTUM achieves logarithmic regret on fixed ?-acutely convex losses, we now generalize to stochastic losses. In order to do this we will necessarily have to make some assumptions about the process generating the stochastic losses. We encapsulate these assumptions in a stochastic version of ?-acute convexity, given below. Definition 6. Suppose for all t, gt is such that E[gt —g1 , . . . gt?1] ? ¿(wt) for some convex function ‘ with minimum at w? . Then we say gt is ?-acutely convex in expectation if: ? ? 2 E[gt] ? (wt ? w) ? ? E[kgt kkwt ? w k] where all expectations are conditioned on g1 , . . . , gt?1 . Using this definition, a fairly straightforward calculation gives us the following result. Theorem 7. Suppose gt is ?-acutely convex in expectation and gt is bounded kgt k ? Lmax with probability 1. Then F REE R EX M OMENTUM achieves expected regret:

Lmax kw? k ? ? [R (w)] ? O E T ? Proof. Throughout this proof, all expectations are conditioned on prior subgradients. By Corollary 2 and Jensen?s inequality we have ” T # ?

X Lmax 2Lmax ? gt ? (wt ? w) ? E 405Lmax + 2Lmax B + 3 ? B log(2BT + 1) E 1 + L1 t=1 v ? ! u T X ? u kw? k2 + kg kkw? ? w k2 log(2T B + 1)(2 + log(T))? +2 5tL max

t

t

t=1

? Lmax 2Lmax ? B log(2BT + 1) ? 405Lmax + 2Lmax B + 3 ? v ! u T X ? u t + 2 5 Lmax kw? k2 + E[kgt kkw? ? wt k2] log(2T B + 1)(2 + log(T)) t=1

? Lmax 2Lmax ? B log(2BT + 1) ? 405Lmax + 2Lmax B + 3 ? v ! u T X ? u 1 + 2 5tLmax kw? k2 + E[gt ? (wt ? w?)] log(2T B + 1)(2 + log(T)) ? t=1 Set R = E

i ? g (w ? w) . Then we have shown t t=1 t s

? R log(2T B + 1)(2 + log(T)) R ? 2 5 Lmax kw? k2 + ? ? Lmax 2Lmax ? + 405Lmax + 2Lmax B + 3 B log(BT + 1) ? s ”

R ? 2 ? =O Lmax kw k + ?

hP T

And now we use Proposition 5 to conclude:

T X Lmax kw? k ? ? E[gt ? (wt ? w)] = O ? t=1 ? hides at most a O(log4 (T)) dependence on T . as desired, where again O Exactly the same argument with an extra factor of d applies to the regret of F REE R EX M OMENTUM with coordinate-wise updates. 6

5

Examples of ?-acute convexity in expectation

In this section, we show that ?-acute convexity in expectation is a condition that arises in practice, justifying the relevance of our logarithmic regret bounds. To do this, we show that a quadratic lower bound on the expected loss implies ?-acute convexity, demonstrating acutely convexity is a weaker condition than strong convexity. Proposition 8. Suppose E[gt —g1 , . . . , gt?1] ? ¿(wt) for some convex ‘ such that for some ? ¿ 0 and w? = argmin ‘, ‘(w) ? ‘(w?) ? ?2 kw ? w? k2 for all w ? W . Suppose kgk ? Lmax with probability 1. Then gt is 2L?max -acutely convex in expectation. Proof. By convexity and the hypothesis of the proposition: E[gt] ? (wt ? w?) ? ‘(wt) ? ‘(w?) ? ? ? ? 2 ? 2 2 kwt ? w k ? 2Lmax E[kgt kkwt ? w k With Proposition 8, we see that F REE R EX M OMENTUM obtains logarithmic regret for any loss that is larger than a quadratic, without requiring knowledge of the parameter ? or the Lipschitz bound Lmax . Further, this result requires only the expected loss ‘ = E[‘t] to have a quadratic lower bound - the individual losses ‘t themselves need not do so. The boundedness of W makes it surprisingly easy to have a quadratic lower bound. Although a quadratic lower bound for a function ‘ is easily implied by strong convexity, the quadratic lower bound is a significantly weaker condition. For example, since W has diameter B, kwk ? B1 kwk2 and so the absolute value is B1 -acutely convex, but not strongly convex. The following Proposition shows that existence of a quadratic lower bound is actually a local condition; so long as the expected loss ‘ has a quadratic lower bound in a neighborhood of w? , it must do so over the entire space W : Proposition 9. Supppose ‘ : W ? R is a convex function such that ‘(w) ? ‘(w?) ? ?2 kw ? w? k

?r ? for all w with kw ? w? k ? r. Then ‘(w) ? ‘(w?) ? min 2B , 2 kw ? w? k2 for all w ? W . ? Proof. We translate by w? to assume without loss of generalityh that w = 0. Then i the statement kwk rw ? is clear for kwk ? r. By convexity, ‘(w) ? ‘(w) ? r ‘ kwk ? ‘(w?) ? ?r 2 kwk ? ?r 2 2B kwk .

Finally, we provide a simple motivating example of an interesting problem we can solve with an ?-acutely convex loss that is not strongly convex: computing the median. Proposition 10. Let W = [a, b], and ‘t (w) = —w ? xt — where each xt is drawn i.i.d. from some fixed distribution with a continuous cumulative distribution function D, and assume D(x?) = 12 . Further, 0 suppose —2D(w) ? 1— ? F —w ? x? — for all —w ? x? — ? G. Suppose gt = ‘t (wt) for wt 6= xt and gt = ?1 with equal probability if wt = xt . Then gt is min

FG b?a , F

-acutely convex in expectation.

Proof. By a little calculation, E[gt] = ‘0 (wt) = 2D(wt) ? 1, and E[—gt —] = 1. Since ‘0 (x?) = 0, w? = x? (the median). For —wt ? x? — ? G, we have —2D(w) ? 1— ? F G, which gives E[gt] ? (wt ? FG w?) ? b?a —](wt ? w?)2 . For —wt ? x? — ? G, we have E[gt] ? (wt ? w?) ? F E[—gt —](wt ? w?)2 , E[—gt

FG so that gt is min b?a , F -acutely convex in expectation. Proposition

10 shows that we can obtain low regret for an interesting stochastic problem without curvature. The condition on the cumulative distribution function D is asking only that there be positive density in a neighborhood of the median; it would be satisfied if D0 (w) ? F for —w— ? G. If the expected loss ' is ?-strongly convex, we can apply Proposition 8 to see that ' is ?/2-aligned, ? max kw? k/?). This is different from the usual regret and then use Theorem 7 to obtain a regret of O(L 2 ? bound of O(Lmax /?) obtained by Online Newton Step [13], which is due to an inefficiency in using the wearker ?-alignment condition. Instead, arguing from the regret bound of Corollary 2 directly, we can recover the optimal regret bound: 7

Corollary 11. Suppose each 't is an independent random variable with E['t] = ' for some ?-strongly convex ' with minimum at w? . Then the expected regret of F REE R EX M OMENTUM satisfies " T # X ? 2 /?) '(wt) ? '(w?) ? O(L E max

t=1

? hides terms that are logarithmic in T B. Where the O Proof. From strong-convexity, we have kwt ? w? k2 ?

2 ('(wt) ? '(w?)) ?

Therefore applying Corollary 2 we have ? ?v " T # u T u X X ? ? ?tL2max E[kwt ? w? k2]? '(wt) ? '(w?) ? O E[RT (w)] = E t=1

t=1

p ? L2 E[RT (w?)]) ? O(max So that applying Proposition 5 we obtain the desired result. As a result of Corollary 11, we see that F REE R EX M OMENTUM obtains logarithmic regret for ?aligned problems and also obtains the optimal (up to log factors) regret bound for ?-strongly-convex problems, all without requiring any knowledge of the parameters ? or ?. This stands in contrast to prior algorithms that adapt to user-supplied curvature information such as Adaptive Gradient Descent [14] or (A, B)-prod [15].

6

Conclusions and Open Problems

? ? We have presented an algorithm, F REE R EX M OMENTUM , that achieves both O(BL T) regret in max

? Lmax B regret in ?-acutely convex stochastic settings without requiring adversarial settings and O ? any prior information about any parameters. We further showed that a quadratic lower bound on the expected loss implies acute convexity, so that while strong-convexity is sufficient for acute convexity, other important loss families such as the absolute loss may also be acutely convex. Since F REE R EX M OMENTUM does not require prior information about any problem parameters, it does not require any hyperparameter tuning to be assured of good convergence. Therefore, the user need not actually know whether a particular problem is adversarial or acutely convex and stochastic, or really much of anything at all about the problem, in order to use F REE R EX M OMENTUM. There are still many interesting open questions in this area. First, we would like to find an efficient way to implement the F REE R EX M OMENTUM algorithm or some variant directly, without appealing to coordinate-wise updates. This would enable us to remove the factor of d we incur by using coordinate-wise updates. Second, our modification to F REE R EX is extremely simple and intuitive, but our analysis makes use of some of the internal logic of F REE R EX. It is possible, however, that any algorithm with sufficiently low regret can be modified in? a similar way to achieve our results. 4 Finally, we?observe that while log (T) is much better than T asymptotically, it turns out that log4 (T) ¿ T for T ¡ 1011 , which casts the practical relevance of our logarithmic bounds in doubt. Therefore we hope that this work serves as a starting point for either new analysis or algorithm design that further simplifies and improves regret bounds.

2 References

[1] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In Proceedings of the 20th International Conference on Machine Learning (ICML-03), pages 928?936, 2003. [2] Shai Shalev-Shwartz. Online learning and online convex optimization. Foundations and Trends in Machine Learning, 4(2):107?194, 2011.

8

[3] Jacob Abernethy, Peter L Bartlett, Alexander Rakhlin, and Ambuj Tewari. Optimal strategies and minimax lower bounds for online convex games. In Proceedings of the nineteenth annual conference on computational learning theory, 2008. [4] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. In Conference on Learning Theory (COLT), 2010. [5] H. Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. In Proceedings of the 23rd Annual Conference on Learning Theory (COLT), 2010. [6] Francesco Orabona and D?vid P?l. Coin betting and parameter-free online learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 577?585. Curran Associates, Inc., 2016. [7] Ashok Cutkosky and Kwabena Boahen. Online learning without prior information. arXiv preprint arXiv:1703.02629, 2017. [8] Tim van Erven and Wouter M Koolen. Metagrad: Multiple learning rates in online learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 3666?3674. Curran Associates, Inc., 2016. [9] Wouter M Koolen, Peter Gr?nwald, and Tim van Erven. Combining adversarial guarantees and stochastic fast rates in online learning. In Advances in Neural Information Processing Systems, pages 4457?4465, 2016. [10] Francesco Orabona. Dimension-free exponentiated gradient. In Advances in Neural Information Processing Systems, pages 1806?1814, 2013. [11] Ashok Cutkosky and Kwabena A Boahen. Online convex optimization with unconstrained domains and losses. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 748?756. Curran Associates, Inc., 2016. [12] Brendan Mcmahan and Matthew Streeter. No-regret algorithms for unconstrained online convex optimization. In Advances in neural information processing sys-

tems, pages 2402?2410, 2012. [13] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. Machine Learning, 69(2):169?192, 2007. [14] Peter L Bartlett, Elad Hazan, and Alexander Rakhlin. Adaptive online gradient descent. In NIPS, volume 20, pages 65?72, 2007. [15] Amir Sani, Gergely Neu, and Alessandro Lazaric. Exploiting easy data in online optimization. In Advances in Neural Information Processing Systems, pages 810?818, 2014.

9