Online Learning with a Hint

Authored by:

Ofer Dekel
Patrick Jaillet
Nika Haghtalab
arthur flajolet

Abstract

We study a variant of online linear optimization where the player receives a hint about the loss function at the beginning of each round. The hint is given in the form of a vector that is weakly correlated with the loss vector on that round. We show that the player can benefit from such a hint if the set of feasible actions is sufficiently round. Specifically, if the set is strongly convex, the hint can be used to guarantee a regret of O(log(T)), and if the set is q-uniformly convex for qin(2,3), the hint can be used to guarantee a regret of o(sqrt{T}). In contrast, we establish Omega(sqrt{T}) lower bounds on regret when the set of feasible actions is a polyhedron.

1 Paper Body

Online linear optimization is a canonical problem in online learning. In this setting, a player attempts to minimize an online adversarial sequence of loss functions while incurring a small regret, compared ? to the best offline solution. Many online algorithms exist that are designed to have ? a regret of O(T) in the worst case and it has been known that one cannot avoid a regret of ?(T) in the worst case. While this worst-case perspective on online linear optimization has lead to elegant algorithms and deep connections to other fields, such as boosting [9, 10] and game theory [4, 2], it can be overly pessimistic. In particular, it does not account for the fact that the player may have ? side-information that allows him to anticipate the upcoming loss functions and evade the ?(T) regret. In this work, we go beyond this worst case analysis and consider online linear optimization when additional information in the form of a function that is correlated with the loss is presented to the player. More formally, online convex optimization [24, 11] is a T -round repeated game between a player and an adversary. On each round, the player chooses an action xt from a convex set of feasible actions K ? Rd and the adversary chooses a convex bounded loss function ft . Both choices are revealed and the player incurs a loss of ft (xt).

The player then uses its knowledge of ft to adjust its strategy for the subsequent rounds. The player?s goal is to accumulate a small loss compared to the best fixed action in hindsight. This value is called regret and is a measure of success of the player?s algorithm. When ? the adversary is restricted to Lipschitz loss functions, several algorithms are known to guarantee O(T) regret [24, 16, 11]. If we further restrict the adversary to strongly convex loss functions, the regret bound improves to O(log(T ?)) [14]. However, when the loss functions are linear, no online algorithm can have a regret of o(T) [5]. In this sense, linear loss functions are the most difficult convex loss functions to handle [24]. 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

In this paper, we focus on the case where the adversary is restricted to linear Lipschitz loss functions. More specifically, we assume that the loss function ft (x) takes the form cTt x, where ct is a bounded loss vector in C ? Rd . We further assume that the player receives a hint before choosing the action on each round. The hint in our setting is a vector that is guaranteed to be weakly correlated with the loss vector. Namely, at the beginning of round t, the player observes a unit-length vector vt ? Rd such that vtT ct ? ?kct k2 , and where ? is a small positive constant. So long as this requirement is met, the hint could be chosen maliciously, possibly by an adversary who knows how the player?s algorithm uses the hint. Our goal is to develop a player strategy that takes these hints into account, and to understand when hints of this type make the problem provably easier and lead to smaller regret. We show that the player?s ability to benefit from the hints depends on the geometry of the player?s action set K. Specifically, we characterize the roundness of the set K using the notion of (C, q)uniform convexity for convex sets. In Section 3, we show that if K is a (C, 2)-uniformly convex set (or in other words, if K is a C-strongly convex set), then we can use the hint to design a player strategy that improves the regret guarantee to O (C?)?1 log(T) , where our O(?) notation hides a polynomial dependence on the dimension d and other constants. Furthermore, as we show in Section 4, if K is a (C, q)-uniformly convex set for q ? (2, 3), we can use the hint to improve the

2?q 1 regret to O (C?) 1?q T 1?q , when the hint belongs to a small set of possible hints at every step. In Section 5, we prove lower bounds on the regret of any online algorithm in this model. We first show that when ? K is a polyhedron, such as a L1 ball, even a stronger form of hint cannot guarantee a regret of o(T). Next, we prove a lower bound of ?(log(T)) regret when K is strongly convex. 1.1

Comparison with Other Notions of Hints

The notion of hint that we introduce in this work generalizes some of the notions of predictability on online learning. Hazan and Megiddo [13] considered as an example a setting where the player knows the first coordinate of the loss vector at all rounds, and showed that when —ct1 — ? ? and when the set of feasible actions is the Euclidean ball, one can achieve a regret of O(1/? ? log(T)). Our work directly improves over this result, as in our setting a hint vt = ?e1 also achieves O(1/? ? log(T)) regret, but we can deal with hints in different directions at different rounds and we allow for general uniformly convex action sets. Rakhlin and Sridharan [20] considered online learning with predictable sequences, with a notion of predictability that is concerned with the gradient of the convex loss functions. They show that qPif the player receives a T 2 hint Mt at round t, then the regret of the algorithm is at most O(t=1 k?ft (xt) ? Mt k?). c?(x) In the case of linear loss functions, this implies that having an estimate vector x c c0t of the loss vector within distance v loss vector ct results in ? ? of the true an improved regret bound of O(? T). In contrast, we consider a notion of c0 hint that pertains to the direction of the loss vectorc rather than its location. v ? Our work shows that merely knowing whether the loss vector positivelyx? or c negatively correlates with another vector is sufficient toxachieve improved c?(x)+? c(y) regret bound, when the set is uniformly convex. That is, rather than having x+y 2 z= access to an approximate value of ct , we only need to2 have access to wa Figure 1: Comparison c?(z) halfspace that classifies ct correctly with a margin. This y notion of hinty?is between notions of hint. weaker that the notion of hint in the work of Rakhlin and Sridharan [20] when the approximation error satisfies kct ? c0t k2 ? ? ? kct k2 for ?v? [0, 1). In this case one can 1 use c0t / kc0t k2 as the direction of the hint in our setting and achieve a regret of O(1?? log T) when the set is strongly convex. This shows that when the set of feasible actions is strongly convex, a directional hint can improve the regret bound beyond what has been known to be achievable by an approximation hint. However, we note that our results require the hints to be always valid, whereas the algorithm of Rakhlin and Sridharan [19] can adapt to the quality of the hints. We discuss these works and other related works, such as [15], in more details in Appendix A.

2

Preliminaries

We begin with a more formal definition of online linear optimization (without hints). Let A denote the player?s algorithm for choosing its actions. On round t the player uses A and all of the information 2

it has observed so far to choose an action xt in a convex compact set K ? Rd . Subsequently, the adversary chooses a loss vector ct in a compact set C ? Rd . The player and the adversary reveal their actions and the player incurs the loss cTt xt . The player?s regret is defined as R(A, c1:T) =

T X t=1

cTt xt ? min x?K

T X

cTt x.

t=1

In online linear optimization with hints, the player observes vt ? Rd with kvt k2 = 1, before choosing xt , with the guarantee that vtT ct ? ?kct k2 , for some ? ¿ 0.

We use uniform convexity to characterize the degree of convexity of the player?s action set K. Informally, uniform convexity requires that the convex combination of any two points x and y on the boundary of K be sufficiently far from the boundary. A formal definition is given below. Definition 2.1 (Pisier [18]). Let K be a convex set that is symmetric around the origin. K and the Banach space defined by K are said to be uniformly convex if for any 0 ¡ ¡ 2 there exists a ? ¿ 0 such that for any pair of points x, y ? K with kxkK ? 1, kykK ? 1, kx ? ykK ? , we have

x+y ? 1 ? ?. The modulus of uniform-convexity ?K () is the best possible ? for that , i.e., 2 K

x + y

?K () = inf 1 ? : kxkK ? 1, kykK ? 1, kx ? ykK ? . 2 K

For brevity, we say that K is (C, q)-uniformly convex when ?K () = Cq and we omit C when it is clear from the context.

Examples of uniformly convex sets include Lp balls for any 1 ¡ p ¡ ? with modulus of convexity ?Lp () = Cp p for p ? 2 and a constant Cp and ?Lp () = (p ? 1)2 for 1 ¡ p ? 2. On the other hand, L1 and L? units balls are not uniformly convex. When ?K () ? ?(2), we say that K is strongly convex. Another notion of convexity we use in this work is called exp-concavity. A function f : K ? R is exp-concave with parameter ? ¿ 0, if exp(??f (x)) is a concave function of x ? K. This is a weaker requirement than strong convexity when the gradient of f is uniformly bounded [14]. The next proposition shows that we can obtain regret bounds of order ?(log(T)) in online convex optimization when the loss functions are exp-concave with parameter ?. Proposition 2.2 ([14]). Consider online convex optimization on a sequence of loss functions f1 , . . . , fT over a feasible set K ? Rd , such that all t, ft : K ? R is exp-concave with parameter ? ¿ 0. There is an algorithm, with runtime polynomial in d, which we call AEXP , with a regret that is at most ?d (1 + log(T + 1)). Throughout this work, we draw intuition from basic orthogonal geometry. Given any vector x and a hint v, we define x v = (xT v)v and x v = x?(xT v)v, as the parallel and the orthogonal components of x with respect to v. When the hint v is clear from the context we simply use x and x to denote these vectors. T

T

Naturally, our regret bounds involve a number of geometric parameters. Since the set of actions of the adversary C is compact, we can find G ? 0 such that kck2 ? G for all c ? C. When K is uniformly convex, we denote K = {w ? Rd — kwkK ? 1}. In this case, since all norms are equivalent in finite dimension, there exist R ¿ 0 and r ¿ 0 such that Br ? K ? BR , where Br (resp. BR) denote the L2 unit ball centered at 0 with radius r (resp. R). This implies that R1 k?k2 ? k?kK ? 1r k?k2 .

3

Improved Regret Bounds for Strongly Convex K

At first sight, it is not immediately clear how one should use the hint. Since vt is guaranteed to satisfy cTt vt ? ?kct k2 , moving the action x in the direction ?vt always decreases the loss. One could hope to get the most benefit out of the hint by choosing xt to be the extremal point in K in the direction ?vt . However, this na?ve strategy could lead to a linear regret in the worst case. For example, say that ct = (1, 12) and vt = (0, 1) for all t and let K be the Euclidean unit ball. Choosing xt = ?vt ?2 ? would incur a loss of ? T2 , while the best fixed action in hindsight, the point (? , ?15), would incur a 5 loss of

? ? 5 2 T.

The player?s regret would therefore be 3

?

5?1 2 T.

T

T

T

w?K:w =? x

x w?K:w =?

T

w?K:w =? x

where the last transition holds by the fact that ct = ct 2 vt since the hint is valid. This provides an intuitive understanding of a measure of convexity x ? of our virtual loss functions. When K is uniformly convex then the x c function c?t (?) demonstrates convexity in the subspace orthogonal to vt . To see that, note that for any x and y that lie in the space c?(x)+? c(y) x+y 2 orthogonal to vt , their mid point x+y transforms to a point that z= w 2 2 is farther away in the direction of ?vt than the midpoint of the c?(z) transformations of x and y. As shown in Figure 3, the modulus y of uniform convexity of K affects the degree of convexity of y? c?t (?). We note, however, that c?t (?) is not strongly convex in v all directions. In fact, c?t (?) is constant in the direction of vt . Nevertheless, the properties shown here allude to the fact that Figure 3: Uniform-convexity of the c?t (?) demonstrates some notion of convexity. As we show in the feasible set affects the convexity the virtual loss function. next lemma, this notion is indeed exp-concavity: ??C?r Lemma 3.1. If K is (C, 2)-uniformly convex, then c?t (?) is 8 G?R2 -exp-concave. ?t (?) = 0 is a Proof. Let ? = 8 ??C?r G?R2 . Without loss of generality, we assume that ct 6= 0, otherwise c constant function and the proof follows immediately. Based on the above discussion, it is not hard to see that c?t (?) is continuous (we prove this in more detail in the Appendix D.1. So, to prove that c?t (?) is exp-concave, it is sufficient to show that

x+y 1 1 exp ?? ? c?t ? exp (?? ? c?t (x)) + exp (?? ? c?t (y)) ?(x, y) ? K. 2 2 2 Consider (x, y) ? K and choose corresponding (? x, y?) ? K such that c?t (x) = cTt x ? and c?t (y) = cTt y?. Without loss of generality, we have k? xkK = k? y kK = 1, as we can always choose corresponding x ?, y? that are extreme points of K. Since exp(??? ct (?)) is decreasing in c?t (?), we have x+y 2

= w

x ?+? y 2

? ?K (k? x ? y?kK) kvvttk

K

vt

=(x+y 2)

(2)

vt

y satisfies kwkK ? 1, since kwkK ? x?+? 2 T
Note that w =
exp(?? ? cTt w).
max
kwkK ?1
vt
vt (x+y . 2) T
exp ?? ? c?t
T
T
v
.
In words, we consider the 1-dimensional subspace spanned by vt and its (d ? 1)-dimensional orthogonal subspace separately. For any c?(x) action x ? K, we find another point, w ? K, that equals x in the x (d ? 1)-dimensional orthogonal subspace, but otherwise incurs the v optimal loss. The value of the virtual loss c?t (x) is defined to be the value of the original loss function ct at w. The virtual loss simulates the process of moving x as far as possible in the direction ?vt without c changing its value in any other direction (see Figure 2). This can be Figure 2: Virtual function c?(?). more formally seen by the following equation.
arg min cTt w = arg min (ct)T x ? + (ct)T w = arg min vtT w, (1) T
y
vt
T
(? z)
z
=x
T
z0
vt
T
xc
w
s.t.
w?K
T
c?t (x) = min cTt w
T
))
Intuitively, the flaw of this na?ve strategy is that the hint does not give the player any information about the (d ? 1)-dimensional subspace orthogonal to vt . Our solution is to use standard online learning machinery to learn how to act in this orthogonal subspace. Specifically, on round t, we use vt to define the following virtual loss function:
K

+

?K (k? x ? y?kK) ? 1 (see also Figure 3). Moreover, w = So, by using this w in Equation (2), we have

x+y ? c T vt exp ?? ? c?t ? exp ? ? (cTt x ? + cTt y?) + ? ? t ? ?K (k? x ? y?kK) . (3) 2 2 kvt kK 4

x

v c

On the other hand, since kvt kK ? 1r kvt k2 = 1r and k? x ? y?kK ? R1 k? x ? y?k2 , we have

T 1 c t vt 2 x ? y?k2 ? ?K (k? x ? y?kK) ? exp ? ? r ? ? ? kct k2 ? C ? 2 ? k? exp ? ? kvt kK R T 2 ! ??C ?r ct x ? cTt y? ? exp ? ? ? kct k2 ? ? R2 kct k2 kct k2

2 T T 2 (?/2) ? (ct x ? ? ct y?) ? exp 2 ? 1 ?

1 ? ? exp ? (cTt x ? ? cTt y?) + ? exp ? (cTt y? ? cTt x ?) , 2 2 2 2 where the penultimate inequality follows by the definition of ? and the last inequality is a consequence of the inequality exp(z 2 /2) ? 21 exp(z) + 12 exp(?z), ?z ? R. Plugging the last inequality into (3) yields

? n ?

? o x+y 1 exp ??? ct ( ) ? exp ? (cTt x (cTt x (cTt y? ? cTt x ? + cTt y?) ? exp ? ? cTt y?) + exp ?) 2 2 2 2 2 1 1 ?) = exp (?? ? cTt y?) + exp (?? ? cTt x 2 2 1 1 = exp (?? ? c?t (y)) + exp (?? ? c?t (x)) , 2 2 which concludes the proof. Now, we use the sequence of virtual loss functions to reduce our problem to a standard online convex optimization problem (without hints). Namely, the player applies AEXP (from Proposition 2.2), which is an online convex optimization algorithm known to have O(log(T)) regret with respect to exp-concave functions, to the sequence of virtual loss functions. Then our algorithm takes the action x ?t ? K that is prescribed by AEXP and moves it as far as possible in the direction of ?vt . This process is formalized in Algorithm 1. Algorithm 1 Ahint FOR S TRONGLY C ONVEX K For t = 1, . . . , T , 1. Use Algorithm AEXP with the history c?? (?) for ? ¡ t, and let x ?t be the chosen action. vt

T

T

2. Let xt = arg minw?K (vtT w) s.t. w

=x ?t

vt

. Play xt and receive ct as feedback.

Next, we show that the regret of algorithm AEXP on the sequence of virtual loss functions is an upper bound on the regret of Algorithm 1. Lemma 3.2. For any sequence of loss functions c1 , . . . , cT , let R(Ahint , c1:T) be the regret of algorithm Ahint on the sequence c1 , . . . , cT , and R(AEXP , c?1:T) be the regret of algorithm AEXP on the sequence of virtual loss functions c?1 , . . . , c?T . Then, R(Ahint , c1:T) ? R(AEXP , c?1:T). T

Proof. Equation (1) provides an equivalent definition xt = arg minw?K (cTt w) s.t. w vt = x ? t vt . Using this, we show that the loss of algorithm Ahint on the sequence c1:T is the same as the loss of algorithm AEXP on the sequence c?1:T . T

min

T

T X

T

xt t=1 w?K:w =?

t=1

cTt w =

T X t=1

cTt (arg min T

c?t (? xt) =

T

T X

cTt w) =

w?K:w =? xt

T X

cTt xt .

t=1

Next, we show that the offline optimal on the sequence c?1:T is more competitive that the offline optimal on the sequence c1:T . First note that for any x and t, c?t (x) = minw?K:w =x cTt w ? cTt x. PT PT Therefore, minx?K t=1 c?t (x) ? minx?K t=1 cTt x. The proof concludes by T

T X t=1

cTt xt ? min x?K

T X t=1

cTt x ?

T X t=1

5

c?t (? xt) ? min x?K

T X t=1

T

R(Ahint , c1:T) =

c?t (x) = R(AEXP , c?1:T).

Our main result follows from the application of Lemmas 3.1 and 3.2. Theorem 3.3. Suppose that K ? Rd is a (C, 2)-uniformly convex set that is symmetric around the origin, and Br ? K ? BR for some r and R. Consider online linear optimization with hints where the cost function at round t is kct k2 ? G and the hint vt is such that cTt vt ? ?kct k2 , while kvt k2 = 1. Algorithm 1 in combination with AEXP has a worst-case regret of d ? G ? R2 R(Ahint , c1:T) ? ? (1 + log(T + 1)). 8? ? C ? r Since AEXP requires the coefficient of exp-concavity to be given as an input, ? needs to be known a priori to be able to use Algorithm 1. However, we can use a standard doubling trick to relax this requirement and derive the same asymptotic regret bound. We defer the presentation of this argument to Appendix B.

4

Improved Regret Bounds for (C, q)-Uniformly Convex K

In this section, we consider any feasible set K that is (C, q)-uniformly convex for q ? 2. Our results differ from the previous section in two aspects. First, our algorithm can be used with (C, q)-uniformly convex feasible sets for any q ? 2 compared to the results of the previous section that only hold for strongly convex sets (q = 2). On the other hand, the approach in this section requires the hints to be restricted to a finite set of vectors V. We show that when K is (C, q)-uniformly convex for q ¿ 2, ? 2?q our regret is O(T 1?q). If q ? (2, 3), this is an improvement over the worst case regret of O(T) guaranteed in the absence of hints. We first consider the scenario where the hint is always pointing in the same direction, i.e. vt = v for some v and all t ? [T]. In this case, we show how one can use a simple algorithm that picks the best performing action so far (a.k.a the Follow-The-Leader algorithm) to obtain improved regret bounds. We then consider the case where the hint belongs to a finite set V. In this case, we instantiate one copy of the Follow-The-Leader algorithm for each v ? V and combine their outcomes in order to obtain improved regret bounds that depend on the cardinality of V, which we denote by —V—. Lemma 4.1. Suppose that vt = v for all t = 1, ? ? ? , T and that K is (C, q)-uniformly convex that is symmetric around the origin, and Br ? K ? BR for some r and R. Consider the algorithm, called Pt P Follow-The-Leader (FTL), that at every round t, plays xt ? arg minx?K ? ¡t cT? x. If ? =1 cT? v ? 0 for all t = 1, ? ? ? , T , then the regret is bounded as follows, !1/(q?1)

1/(q?1) X T q kvkK ? Rq kct k2 R(AFTL , c1:T) ? . ? Pt T 2C ? =1 c? v t=1 Furthermore, when v is a valid hint with margin ?, i.e., cTt v ? ? ? kct k2 for all t = 1, ? ? ? , T , the right-hand side can be further simplified to obtain the regret bound: 1 R(AFTL , c1:T) ? ? G ? (ln(T) + 1) if q = 2 2? and q?2 1 q?1 R(AFTL , c1:T) ? ? T q?1 if q ¿ 2, ?G? 1/(q?1) q?2 (2?) where ? = kvkC???Rq . K

Proof. We use a well-known inequality, known as FT(R)L Lemma (see e.g., [12, 17]), on the regret incurred by the FTL algorithm: R(AFTL , c1:T) ?

T X t=1

cTt (xt ? xt+1).

Without loss of generality, we can assume that kxt kK = kxt+1 kK = 1 since the maximum of a linear function is attained at a boundary point. Since K is (C, q)-uniformly convex, we have

xt + xt+1

? 1 ? ?K (kxt ? xt+1 k). K

2 K

6

This implies that

xt + xt+1 v

? 1.

? ? (kx ? x k) K t t+1 K

2 kvkK K Pt Moreover, xt+1 ? arg minx?K xT ? =1 c? . So, we have !T

t t t X X X xt + xt+1 v c? c? = xTt+1 c? . ? ?K (kxt ? xt+1 kK) ? inf xT x?K 2 kvkK ? =1 ? =1 ? =1 Pt Rearranging this last inequality and using the fact that ? =1 v T c? ? 0, we obtain: !T ! Pt

t t q T X X C ? kxt ? xt+1 k2 xt ? xt+1 T ? =1 v c? c? ? ? ?K (kxt ? xt+1 kK) ? ? v c? . 2 kvkK kvkK ? Rq ? =1 ? =1 Pt?1 By definition of FTL, we have xt ? arg minx?K xT ? =1 c? , which implies: !T t?1 X xt+1 ? xt c? ? 0. 2 ? =1 Summing up the last two inequalities and setting ? = kvkC???Rq , we derive: K ! !

t t X X ? (cT (xt ? xt+1))q x ? x ? t t+1 q . cTt v T c? ? kxt ? xt+1 k2 ? ? v T c? ? t ? ? q 2 ? ? kct k2 ? =1 ? =1 Pt Rearranging this last inequality and using the fact that ? =1 v T c? ? 0, we obtain: !1/(q?1) q kct k2 1 T ? Pt —ct (xt ? xt+1)— ? . (4) T (2?/?)1/(q?1) ? =1 v c? Summing (4) over all t completes the proof of the first claim. The regret bounds for when v T ct ? ? ? kct k2 for all t = 1, ? ? ? , T follow from the first regret bound. We defer this part of the proof to Appendix D.2. Note that the regret bounds become O(T) when q ? ?. This is expected because Lq balls are q-uniformly convex for q ? 2 and converge to L? balls as q ? ? and it is well-known that Follow-The-Leader yields ?(T) regret in online linear optimization when K is a L? ball. Using the above lemma, we introduce an algorithm for online linear optimization with hints that belong to a set V. In this algorithm, we instantiate one copy of the FTL algorithm for each possible direction of the hint. On round t, we invoke the copy of the algorithm that corresponds to the direction of the hint vt , using the history of the game for rounds with hints in that direction. We show that the overall regret of this algorithm is no larger than the sum of the regrets of the individual copies. Algorithm 2 Aset : S ET- OF -H INTS For all v ? V, let Tv = ?. For t = 1, . . . , T , P 1. Play xt ? arg minx?K ? ?Tv cT? x and receive ct as feedback. t

2. Update Tvt ? Tvt ? {t}. Theorem 4.2. Suppose that K ? Rd is a (C, q)-uniformly convex set that is symmetric around the origin, and Br ? K ? BR for some r and R. Consider online linear optimization with hints where the cost function at round t is kct k2 ? G and the hint vt comes from a finite set V and is such that cTt vt ? ?kct k2 , while kvt k2 = 1. Algorithm 2 has a worst-case regret of R(Aset , c1:T) ? —V— ? and

R(Aset , c1:T) ? —V— ?

R2 ? G ? (ln(T) + 1), 2C ? ? ? r

Rq 2C ? ? ? r

1/(q?1) ?G? 7

if q = 2,

q?2 q?1 ? T q?1 q?2

if q ¿ 2.

Proof. We decompose the regret as follows: R(Aset , c1:T) =

T X t=1

ct xt ? inf T

x?K

T X t=1

(ct x ? T

)

X

X
v?V
t?Tv
ct xt ? inf T
x?K
X
T
ct x
t?Tv
? —V— ? max R(AFTL , cTv). v?V

The proof follows by applying Lemma 4.1 and by using kvt kK ? (1/r) ? kvt k2 = 1/r. Note that Aset does not require ? or V to be known a priori, as it can compile the set of hint directions as it sees new ones. Moreover, if the hints are not limited to finite set V a priori, then the algorithm can first discretize the L2 unit ball with an ?/2-net and approximate any given hint with one of the hints in the discretized set. Using this discretization technique, Theorem 4.2 can be extended to the setting where the hints are not constrained to a finite set while having a regret that is linear in the size of the ?/2-net (exponential in the dimension d.) Extensions of Theorem 4.2 are discussed in more details in the Appendix C.

5 Lower Bounds

The regret bounds derived in Sections 3 and 4 suggest that the ? curvature of K can make up for the lack of curvature of the loss function to get rates faster than O(T) in online convex optimization, provided we receive additional information about the next move of the adversary in the form of a hint. In this section, we show that the curvature of the player?s decision set K is necessary to get rates ? better than O(T), even in the presence of a hint. As an example, consider the unit cube, i.e. K = {x — kxk? ? 1}. Note that this set is not uniformly convex. Since, the ith coordinate of points in such a set, namely xi , has no effect on the range of acceptable values for the other coordinates, revealing one coordinate does not give us any information about the other coordinates xj for j 6= i. For example, suppose that ct has each of its first two coordinates set to +1 or ?1 with equal probability and all other coordinates set to 1. In this case, even after observing the last d ? 2 coordinates of the loss vector, the problem is reduced to a standard online linear optimization problem in the 2-dimensional unit cube. This choice of ct is known to ? incur a regret of ?(T) [1]. Therefore, online linear optimization with the set K = {x — kxk? ? 1}, ? even in the presence of hints, has a worst-case regret of ?(T). As it turns out, this result holds for any polyhedral set of actions. We prove this by means of a reduction to the lower bounds established in [8] that apply to the online convex optimization framework (without hint). We defer the proof to the Appendix D.4. Theorem 5.1. If the set of feasible actions is a polyhedron then, depending on the set C, either there exists ? a trivial algorithm that achieves zero regret or every online algorithm has worst-case regret ?(T). This is true even if the adversary is restricted to pick a fixed hint vt = v for all t = 1, ? ? ? , T . At first sight, this result may come as a surprise. After all, since any Lp ball with 1 ¡ p ? 2 is strongly convex, one can hope to use a L1+? unit ball K0 to approximate K when K is a L1 ball (which is a polyhedron) and apply the results of Section 3 to achieve better regret bounds. The problem with this approach is that the constant in the modulus of convexity of K0 deteriorates when p ? 1 since ?Lp () = (p ? 1) ? 2 , see [3]. As a result, the regret bound established in Theorem 3.3 1 becomes O(p?1 ? log T). Since the best approximation of a L1 unit ball using a Lp ball is of the 1

form {x ? Rd — d1? p kxkp ? 1}, the distance between the offline benchmark in the definition 1 of regret when using K0 instead of K can be as large as (1 ? d p ?1) ? T , which translates into an 1 additive term of order (1 ? d p ?1) ? T in the regret bound when using K0 as a proxy for K. Due to the inverse dependence of the regret bound obtained in Theorem 3.3 on p ? 1, the optimal choice of ? ? ?1) leads to a regret of order O(? T). p = 1 + O(T

Finally, we conclude with a result that suggests that O(log(T)) is, in fact, the optimal achievable regret when K is strongly convex in online linear optimization with a hint. We defer the proof to the Appendix D.4. 8

Theorem 5.2. If K is a L2 ball then, depending on the set C, either there exists a trivial algorithm that achieves zero regret or every online algorithm has worst-case regret ?(log(T)). This is true even if the adversary is restricted to pick a fixed hint vt = v for all t = 1, ? ? ? , T .

6

Directions for Future Research

We conjecture that the dependence of our regret bounds with respect to T is suboptimal when K is ? (C, q)-uniformly convex for q ¿ 2. We expect the optimal rate to converge to T when ? q ? ? as Lq balls converge to L? balls and it is well known that the minimax regret scales as T in online linear optimization without hints when the decision set is a L? ball. However, this calls for the development of an algorithm that is not based on a reduction to the Follow-The-Leader algorithm, as discussed after Lemma 4.1. We also conjecture that it is possible to relax the assumption that there are finitely many hints when K is (C, q)-uniformly convex with q ¿ 2 without incurring an exponential dependence of the regret bounds (and the runtime) on the dimension d, see Appendix C. Again, this calls for the development of an algorithm that is not based on a reduction to the Follow-The-Leader algorithm. A solution that would alleviate the two aforementioned shortcomings would likely be derived through a reduction to online convex optimization with convex functions that are (C, q)-uniformly convex, for q ? 2, in all but one direction and constant in the other, in a similar fashion as done in Section 3 when q = 2. There has been progress in this direction in the literature, but, to the best of our knowledge, no conclusive result yet. For instance, Vovk [23] studies a related problem but restricts the study to the squared loss function. It is not clear if the setting studied in this paper can be reduced to the setting of square loss function. Another example is given by [21], where the authors consider online convex optimization with general (C, q)-uniformly convex functions in Banach spaces (with no hint) achieving a regret of order O(T (q?2)/(q?1)). Note that this rate matches the one derived in Theorem 4.2. However, as noted above, our setting cannot be reduced to theirs because our virtual loss functions are not uniformly convex in every direction. Acknowledgments Haghtalab was partially funded by an IBM Ph.D. fellowship and a Microsoft Ph.D. fellowship. Jaillet acknowledges the research support of the Office of Naval Research (ONR) grant N00014-15-1-2083. This work was partially done when Haghtalab was an intern at Microsoft Research, Redmond WA.

2 References

[1] Jacob Abernethy, Peter L Bartlett, Alexander Rakhlin, and Ambuj Tewari. Optimal strategies and minimax lower bounds for online convex games. In Proceedings of the 21st Conference on Learning Theory (COLT), pages 415?424, 2008. [2] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. Theory of Computing, 8(1):121?164, 2012. [3] Keith Ball, Eric A Carlen, and Elliott H Lieb. Sharp uniform convexity and smoothness inequalities for trace norms. Inventiones mathematicae, 115(1):463?482, 1994. [4] Avrim Blum and Yishay Monsour. Learning, regret minimization, and equilibria. In Algorithmic Game Theory, pages 79?102. 2007. [5] Nicolo Cesa-Bianchi and G?bor Lugosi. Prediction, learning, and games. Cambridge university press, 2006. [6] Chao-Kai Chiang and Chi-Jen Lu. Online learning with queries. In Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 616?629, 2010. [7] Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In Proceedings of the 25th Conference on Learning Theory (COLT), pages 6?1, 2012. [8] Arthur Flajolet and Patrick Jaillet. No-regret learnability for piecewise linear losses. arXiv preprint arXiv:1411.5649, 2014.

9

[9] Yoav Freund. Boosting a weak learning algorithm by majority. Information and computation, 121(2): 256?285, 1995. [10] Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1):119?139, 1997. [11] Elad Hazan. The convex optimization approach to regret minimization. Optimization for machine learning, pages 287?303, 2012. [12] Elad Hazan and Satyen Kale. Extracting certainty from uncertainty: Regret bounded by variation in costs. In Proceedings of the 23th Conference on Learning Theory (COLT), 2008. [13] Elad Hazan and Nimrod Megiddo. Online learning with prior knowledge. In Proceedings of the 20th Conference on Learning Theory (COLT), pages 499?513, 2007. [14] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. Machine Learning, 69(2-3):169?192, 2007. [15] Ruitong Huang, Tor Lattimore, Andr?s Gy?rgy, and Csaba Szepesv?ri. Following the leader and fast rates in linear prediction: Curved constraint sets and other regularities. In Proceedings of the 30th Annual Conference on Neural Information Processing Systems (NIPS),

pages 4970?4978, 2016. [16] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. Journal of Computer and System Sciences, 71(3):291?307, 2005. [17] H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. Journal of Machine Learning Research, 18:1?50, 2017. [18] Gilles Pisier. Martingales in banach spaces (in connection with type and cotype). Manuscript., Course IHP, Feb, pages 2?8, 2011. [19] Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In Proceedings of the 25th Conference on Learning Theory (COLT), pages 993?1019, 2013. [20] Alexander Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS), pages 3066?3074, 2013. [21] Karthik Sridharan and Ambuj Tewari. Convex games in banach spaces. In Proceedings of the 23rd Conference on Learning Theory (COLT), pages 1?13, 2010. [22] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Compressed Sensing: Theory and Applications, pages 210?268. Cambridge University Press, 2012. [23] Vladimir Vovk. Competing with wild prediction rules. Machine Learning, 69(2-3):193?212, 2007. [24] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In Proceedings of the 20th International Conference on Machine Learning (ICML), pages 928?936, 2003.

10