No Internal Regret with Non-convex Loss Functions

Dravyansh Sharma

Carnegie Mellon University
dravyans@cs.cmu.edu

Abstract

Internal regret is a measure of performance of an online learning algorithm, which measures the change in performance by substituting every occurrence of a given action i by an alternative action j. Algorithms for minimizing internal regret are known for the finite experts setting, including a general reduction to the problem of minimizing external regret for this case. The reduction however crucially depends on the finiteness of the action space. In this work we approach the problem of minimizing internal regret for a continuous action space. For the full information setting, we show how to obtain $\tilde{O}(\sqrt{T})$ internal regret for the class of Lipschitz functions, as well as non-Lipschitz dispersed functions, i.e. the non-Lipschitzness may not concentrate in a small region of the action space. We also consider extensions to partial feedback settings, and again obtain sublinear internal regret. Finally we discuss applications of internal regret minimization over continuous spaces to correlated equilibria in pricing problems and auction design, as well as to data-driven hyperparameter tuning.

Introduction

We consider the problem of repeatedly making decisions in an uncertain environment, using the standard online learning framework. The algorithm or learner has a continuous space $\mathcal{C}$ of actions, and the following game is played for T rounds. At each round or time step t, the learner chooses an action (possibly probabilistically), the environment makes its "move" by choosing a loss function over the space of actions, and the learner then incurs the loss for its action chosen. The most popular metric for measuring the performance of the algorithm is computing its (external, as it is unrelated to learner's choices) 'regret' with respect to the best fixed action from $\mathcal{C}$ played across all rounds, in hindsight. However, this may not be useful in common cases where no single fixed action works well over the entire course of the algorithm's interaction with its environment

One alternative approach to determine if the learner performed well in choosing the actions is to compute the 'regret' of substituting the chosen actions with alternative actions. This gives rise to the notions of internal regret or swap regret (Cesa-Bianchi and Lugosi 2003; Stoltz 2005; Blum and Mansour 2007). While the external regret has been studied in a wide variety of general settings owing to its popularity, the study of internal and swap regret has largely been limited to the case of finite action space, also known as *finite experts* setting. In this work, we consider the problem of minimizing internal regret in the presence of continuously many infinite experts, and non-convex loss functions, in both *full information* (i.e., learner observes loss for all experts) and *partial feedback* (learner only observes their incurred loss) settings. We will particularly focus on the case of Lipschitz losses and piecewise-constant losses motivated by known applications. Internal regret is closely connected to correlated equilibria in multi-player repeated games (Foster and Vohra 1999; Blum and Mansour 2011).

Related Work

Internal regret. The notion of internal regret was first introduced by (Foster and Vohra 1998) in the context of calibrated forecasting. Several low internal regret algorithms have been developed, from initial algorithms with convergence guarantees without regret upper bounds (Foster and Vohra 1999; Hart and Mas-Colell 2000), to general potential-based algorithms with low regret bounds (Cesa-Bianchi and Lugosi 2003), as well as algorithms based on reduction to external regret (Blum and Mansour 2007; Stoltz and Lugosi 2005, 2007). Note that the algorithms, as well as reductions due to (Stoltz 2005; Blum and Mansour 2007) to external regret, were obtained for the finite experts setting. A lower bound of $\Omega(\sqrt{NT})$ on a related notion called *swap regret* for any randomized algorithm was given by (Blum and Mansour 2007) and further improved in the recent work of (Ito 2020) to $\Omega(\sqrt{NT \log N})$.

Bandit setting. For the bandit setting (Stoltz 2005) gave an algorithm with $O(N\sqrt{T \log N})$ regret which runs in exponential time. (Blum and Mansour 2007) obtained a polynomial time algorithm with slightly worse regret bound of $O(N\sqrt{NT \log N})$ via a reduction argument. Recent work of (Ito 2020) modifies their approach to give a polynomial time algorithm with $O(N\sqrt{T})$ regret bound, resulting in a more efficient reduction to external regret without needing first-order bounds.

Other related notions of regret. (Mohri and Yang 2014) consider a generalization of swap regret called *conditional swap regret*, which considers all possible modifications of

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

a learner's action sequence that depend on some fixed bounded history. They further generalize it to a notion of *transductive regret* in (Mohri and Yang 2017), which is a generalization of (1) external regret; (2) internal regret; (3) swap regret; and (4) conditional swap regret. *Sleeping experts* is another popular setting introduced by (Freund et al. 1997) where the set of actions that are available to the decision algorithm varies over time. Algorithms with low regret are known for this problem (Blum and Mansour 2007; Kleinberg, Niculescu-Mizil, and Sharma 2010) and have applications to calendar tracking (Blum 1997), text-categorisation (Cohen and Singer 1999), formulating web-search engine queries (Cohen and Singer 1996) and subgroup fairness (Blum and Lykouris 2020). A related notion is *wide-range regret* (Lehrer 2003; Blum and Mansour 2007; Mohri and Yang 2017).

Correlated equilibrium. There is a tight connection between swap regret and correlated equilibrium (Aumann 1974). For a general-sum game with any finite number of players, if every player has zero internal regret playing a distribution Q over the joint action space then it is a correlated equilibrium. In a repeated game setting, if each player uses an action selection algorithm whose swap regret of this form is sublinear in T, then the empirical distribution of the players actions converges to a correlated equilibrium (Hart and Mas-Colell 2000), and in fact, the benefit of a deviation from a correlated equilibrium is bounded exactly by R/T, where R is the largest swap regret of any of the players.

Data-driven algorithm design. (Gupta and Roughgarden 2016) define a formal learning framework for selecting algorithms from a family of heuristics or setting hyperparameters. It has been successfully applied to parameter tuning several combinatorial problems like integer programming, clustering and low-rank approximation(Balcan, Dick, and Vitercik 2018; Bartlett, Indyk, and Wagner 2022). Prior work on online data-driven parameter selection has focused on external regret against a fixed or dynamic choice of actions (Balcan, Dick, and Vitercik 2018; Sharma, Balcan, and Dick 2020). Our work obtains first internal regret bounds of online parameter configuration.

Summary of Results

We define internal regret on a continuum as a limit. We obtain no internal regret algorithms in the presence of non-convex loss functions. Specifically,

- In the full information setting, for the case of L-Lipschitz loss functions, we obtain an algorithm which achieves regret $O(\sqrt{dT \log RLT})$, where d is the dimension of the Euclidean action space and R is the diameter of action space. Further, for the case of one-dimensional piecewise constant functions, we obtain an algorithm which achieves regret $O(\sqrt{T \log KT})$ under a mild smoothness assumption, where K is a bound on the number of pieces in each loss function. We show that our bounds are near-optimal by providing a $\Omega(\sqrt{T})$ lower bound on the internal regret for our loss functions.
- We extend our results to partial feedback setting, and again obtain sublinear regret $\tilde{O}(T^{\frac{d+1}{d+2}})$, where the soft-O notation suppresses factors other than T as well as logarithmic terms in T.
- We provide applications of our results to designing strategies for achieving correlated equilibrium in multiplayer repeated games, and to data-driven hyperparameter tuning via concrete instantiation for a combinatorial parameter selection problem.

Notation and Terminology

We assume an adversarial online learning model with a continuum of available actions given by a closed and compact set $\mathcal{C}$. At each time step t, an online learner (or algorithm) A selects a distribution p_t over the action space $\mathcal{C}$. After that, the environment (or adversary) selects a loss function $l_t : \mathcal{C} \to [0, 1]$, where t $l_t(a)$ is the loss of the action $a \in \mathcal{C}$ at time t. In the full information setting, the online algorithm receives the complete loss function l_t and experiences a loss $l_t^A = \int_{\mathcal{C}} l_t(a)p_t(a)da$. In the partial information (or bandit) setting, the online algorithm receives loss $l_t(a_t)$, where a_t is distributed according to p_t. The loss of the action a during the first T time steps is $L_T(a) = \sum_{t=1}^{T} l_t(a)$, and the loss of learner A is $L_T^A = \sum_{t=1}^{T} l_t^A$. The goal for the external regret setting is to design an online algorithm that will be close to performance of the best fixed action, that is, to have a loss close to $L_T^* = \min_{a \in \mathcal{C}} L_T(a)$. Formally, one wants to minimize external regret given by $R_T = L_T^A - L_T^*$.

We define the internal regret on continuous action spaces as follows. We assume the action space $\mathcal{C}$ defines a metric space over some norm $||.||$. For $\epsilon > 0$ and actions $a, b \in \mathcal{C}$, let $p_t^{(a,b,\epsilon)}$ be the distribution on $\mathcal{C}$ formed by removing the probability mass in some ball $\mathbf{B}(a, \epsilon)$ and adding it uniformly to points in $\mathbf{B}(b, \epsilon)$, i.e.

$$p_t^{(a,b,\epsilon)}(x) = \begin{cases} 0 & \text{if} x \in \mathbf{B}(a,\epsilon), a \notin \mathbf{B}(b,\epsilon), \\ p_t(x) + p_t^{a \to b}(x) & \text{if } x \in \mathbf{B}(b,\epsilon), a \notin \mathbf{B}(b,\epsilon), \\ p_t(x) & \text{otherwise,} \end{cases}$$

where $p_t^{a \to b}(x) := \frac{\int_{\mathbf{B}(a,\epsilon)} p_t(y)dy}{\mathsf{vol}(\mathbf{B}(b,\epsilon))}$ and $\mathsf{vol}(\mathbf{B}(b, \epsilon)) = \int_{\mathbf{B}(b,\epsilon)} dy$. For technical simplicity, we define $p_t^{(a,b,\epsilon)}(x) = p_t(x)$ for all $x \in \mathcal{C}$ if $a \in \mathbf{B}(b, \epsilon)$ (i.e. swapping with 'almost' the same action does not change the distribution). Then the internal regret is defined as

$$R_T^i = \max_{a,b \in \mathcal{C}} \lim_{\epsilon \to 0} \frac{\sum_{t=1}^{T} \int_{\mathcal{C}} (p_t(x) - p_t^{(a,b,\epsilon)}(x)) l_t(x) dx}{\epsilon},$$

where $l_t : \mathcal{C} \to [0, 1]$ is the loss function at time t. We illustrate this definition with an example.

Lemma 1. *If p_t is a discrete distribution over a finite subset $\mathcal{C}' \subset \mathcal{C}$ for all $t \in [T]$, then*

$$R_T^i = \max_{b \in \mathcal{C}} \hat{R}_T^i(\mathcal{C}' \cup \{b\}),$$

where $\hat{R}_T^i(\mathcal{A})$ denotes the standard notion of internal regret over the finite experts in $\mathcal{A}$ (Blum and Mansour 2011).

Proof. Since $\mathcal{C}'$ is finite, we have $\epsilon_0 = \frac{1}{2}\min_{a\neq b; a,b\in\mathcal{C}'} ||a - b|| > 0$. Clearly, for any $a \in \mathcal{C}'$ and $0 < \epsilon < \epsilon_0$, the ball $\mathbf{B}(a, \epsilon)$ does not contain $b \in \mathcal{C}'$ for $b \neq a$. Thus, $\int_{\mathbf{B}(a,\epsilon)} p_t(y)dy = p_t(a)$ if $a \in \mathcal{C}'$.

As above, let $0 < \epsilon < \epsilon_0$. For $a, b \in \mathcal{C}$, if $a \in \mathbf{B}(b, \epsilon)$ we have $R^i_T = 0$ by definition. We assume $a \notin \mathbf{B}(b, \epsilon)$. If $\mathbf{B}(a, \epsilon) \cap \mathcal{C}' = \{\}$, it is easy to see that $p_t^{(a,b,\epsilon)}(x) = p_t(x)$ for all $x \in \mathcal{C}$ and again $R^i_T = 0$ (since p_t is assumed to have no mass outside of $\mathcal{C}'$). But this will hold for sufficiently small ϵ unless $a \in \mathcal{C}'$. In this case, intuitively, the probability mass on a is uniformly redistributed to points in $\mathbf{B}(b, \epsilon)$ with the mass effectively concentrating at b as $\epsilon \to 0$. Specifically, the limit corresponds to a discrete distribution over $\mathcal{C}' \cup \{b\}$ and the result follows from the definitions. □

We use $[n]$ to denote the set $\{1, \ldots, n\}$. $\mathbf{I}[\cdot]$ will be used to denote the 0-1 valued indicator function.

A Potential Function Analysis

In this section, we will describe the potential function based approach due to Cesa-Bianchi and Lugosi (2003) for finite experts. The problem is parametrized by a decision space $\mathcal{X}$, by an outcome space $\mathcal{Y}$, and by a convex and twice differentiable *potential function* $\phi : \mathbb{R}^N \to \mathbb{R}^+$. At each step $t = 1, 2, \ldots$, the current *state* is represented by a point $R_{t-1} \in \mathbb{R}^N$, where $R_0 = 0^N$. The decision maker observes a vector-valued *drift function* $r_t : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^N$ and selects an element x_t from the decision space $\mathcal{X}$. In return, an outcome $y_t \in \mathcal{Y}$ is received, and the new state of the problem is the “drifted point” $R_t = R_{t-1} + r_t(x_t, y_t)$. The goal of the decision maker is to minimize the potential $\phi(R_t)$ for a given t (which might be known or unknown to the decision maker).

To illustrate the above abstraction, we consider the standard online learning with external regret and express it in the above framework. Here, the decision maker is a learner whose goal is to forecast a hidden sequence $y_1, y_2, \ldots$ of elements in the outcome space $\mathcal{Y}$. At each time t, the learner computes its guess $x_t \in \mathcal{X}$ for the next outcome y_t. This guess is based on the advice $f_{1,t}, \ldots, f_{N,t} \in \mathcal{X}$ of N experts from a fixed pool. The guesses of the learner and the experts are then individually scored using a loss function $l : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. The learner's goal is to keep as small as possible the cumulative regret with respect to each expert, which can be easily modeled within the above abstract decision problem by associating a coordinate to each expert and by defining the components $r_{i,t}$ of the drift function r_t by $r_{i,t}(x_t, y_t) = l(x_t, y_t) - l(f_{i,t}, y_t)$ for $i = 1, ..., N$. The role of the potential function in this framework is to provide a generalized way to measure the size (or distance from the origin) of the cumulative regret which is measured by the state R_t. To obtain the general results, two assumptions are needed on the potential function

Assumption 1. *(Generalized Blackwell's condition). At each time t, a decision $x_t \in \mathcal{X}$ exists such that*

$$\sup_{y_t \in \mathcal{Y}} \nabla(R_{t-1}) \cdot r_t(x_t, y_t) \leq 0$$

Strategies satisfying the above condition tend to keep the point R_t as close as possible to the minimum of the potential by forcing the drift vector to point away from the gradient of the current potential. This gradient descent approach to sequential decision problems is not new, and the prominent eponymous example of a decision strategy of this type is the one used by Blackwell to prove his celebrated approachability theorem (Blackwell 1956), generalizing von Neumann's celebrated minimax theorem to vector-valued payoffs. We also need a second assumption about additivity of the potential function.

Assumption 2. *The potential ϕ can be written as $\phi(u) = \sum_i \phi_i(u_i)$ for all $u = (u_1, ..., u_N) \in \mathbb{R}^N$, where $\phi_i : \mathbb{R} \to \mathbb{R}^+$ is a nonnegative function of one variable. Typically, ϕ_i will be monotonically increasing and convex on $\mathbb{R}$.*

Under these assumptions, the following general theorem is given by (Cesa-Bianchi and Lugosi 2003).

Theorem 2. *Let ϕ be a twice differentiable additive potential function and let $r_1, r_2, ..., r_N \in \mathbb{R}^N$ be such that $\nabla\phi(R_{t-1}) \cdot r_t \leq 0$ for all $t \geq 1$, where $R_t = r_1 + \cdots + r_t$. Let $f : \mathbb{R}^+ \to \mathbb{R}^+$ be an increasing, concave, and twice differentiable auxiliary function such that, for all $t = 1, 2, ...$,*

$$\sup_{u \in \mathbb{R}^N} f(\phi(u)) \sum_{i=1}^{N} \phi_i''(u_i) r_{i,t}^2 \leq C(r_t)$$

for some non-negative function $C : \mathbb{R}^N \to \mathbb{R}^+$. Then, for all $t = 1, 2, ...$,

$$f(\phi(R_t)) \leq f(\phi(0)) + \frac{1}{2}\sum_{s=1}^{t} C(r_s).$$

In order to apply the above theorem, one must choose a potential function ϕ_i and functions f and C. For minimizing the internal regret, one choice is to have the exponential potential function $\phi_i(u) = e^{\eta u}$, by choosing $f(x) = \frac{1}{\eta}\ln x$ and setting $C(r_t) = \eta \max_i r_{i,t}^2$. (Cesa-Bianchi and Lugosi 2003) (Theorem 3) show the existence of a randomized learner which satisfies the Blackwell condition and, by setting the step size parameter $\eta = \sqrt{4 \ln N/T}$ they conclude that the internal regret R^i_T satisfies $R^i_T \leq 2\sqrt{T \ln N}$.

No Internal Regret on a Continuum

We first discuss challenges in extending known approaches for finite experts, i.e. if action space $\mathcal{C} = \{1, 2, \ldots, N\}$ instead of a closed compact set. The key idea of the reduction based approach of (Blum and Mansour 2007) is to design a meta-algorithm which runs N ‘base’ external regret algorithms and has a meta-distribution p over the distributions over actions Q output by the base algorithms, such that p can be computed as the stationary distribution of a finite Markov chain corresponding to matrix Q. While approaches are known to approximate the stationary distribution of a countably infinite Markov chain (Seneta 1980), it is not clear if this approach could be extended to the uncountably infinite setting that we consider.

Moreover the regret bounds obtained in (Blum and Mansour 2007) correspond to the stronger notion of swap regret, but obtain a weaker regret bound which is polynomial in N. This is reasonable for small N, but handling the case of large N efficiently is noted as an open problem in (Blum and Mansour 2011). The above approach of (Cesa-Bianchi and Lugosi 2003) provides an computationally inefficient alternative, and bounds the weaker notion of internal regret.

Lipschitz-continuous Loss Functions

We first consider the case when the loss function $l_t : \mathcal{C} \to [0, 1]$ is a L-Lipschitz continuous function. We will assume that the action space $\mathcal{C} \subset \mathbb{R}^d$ is closed compact and Euclidean. The key idea is to discretize the action space, i.e. obtain a cover for it using sufficiently many experts such that any point in the action space is ϵ-close to some point in the cover. We then apply the results of (Cesa-Bianchi and Lugosi 2003) to obtain algorithms with low internal regret over this large but finite cover. To formalize things, we will use the following well-known discretization lemma.

Lemma 3. *Let $C \subseteq \mathbb{R}^d$ be contained in a ball of radius R. Then the greedy procedure of adding points until there is no longer a point more than ϵ away from all points in the collection obtains an ϵ-cover $C_\epsilon \subset C$ such that $|C_\epsilon| \leq (3R/\epsilon)^d$ and for every $a \in C$ there exists $a' \in C_\epsilon$ such that $||a - a'||_2 \leq \epsilon$.*

Proof. The greedy procedure terminates after adding at most $(3R/\epsilon)^d$ points by a standard covering argument. □

Now our main strategy is to apply the algorithm of (Cesa-Bianchi and Lugosi 2003) over an ϵ-cover $\mathcal{C}_\epsilon \subset \mathcal{C}$ for the action space $\mathcal{C}$. Crucially the logarithmic dependence on the regret bound allows us to choose sufficiently small ϵ (even as small as $1/T$) so as to not incur significant approximation regret for the Lipschitz function. The algorithm, called Discretized Exponential Potential Minimizer (DEPoM) is formally as Algorithm 1.

Our main result in this section is the following theorem about the performance of Algorithm 1.

Theorem 4. *Consider an online learner over action space $\mathcal{C} \subset \mathbb{R}^d$ such that $\mathcal{C}$ is compact, closed and contained in a ball of radius R. If the learner is faced with a sequence of L-Lipschitz losses $l_1, \dots, l_T : \mathcal{C} \to [0, 1]$ and the learner uses Algorithm 1 with $\eta = \sqrt{\frac{4d}{T} \ln 3RLT}$, $\phi(x) = e^{\eta x}$ and with $\epsilon = \frac{1}{LT}$, then the expected internal regret of the learner is $O\left(\sqrt{dT \ln(RLT)}\right)$.*

Proof. By Lemma 3, the greedy procedure of Algorithm 1 uses a ϵ-cover $\mathcal{C}_\epsilon$ with $|\mathcal{C}_\epsilon| \leq (3R/\epsilon)^d$. Now by Corollary 8 of (Cesa-Bianchi and Lugosi 2003), the internal regret of Algorithm 1 when substituting the actions over the ϵ-cover is at most $2\sqrt{T \ln N}$. Substituting $N \leq (3R/\epsilon)^d$, we get that the internal regret w.r.t. to $\mathcal{C}_\epsilon$ is at most

$$2\sqrt{T \ln\left((3R/\epsilon)^d\right)} = O\left(\sqrt{dT \ln \frac{R}{\epsilon}}\right).$$

Algorithm 1: DEPoM(η, ϵ)

```
Input: step size $\eta \in [0, 1]$, discretization parameter $\epsilon$.
Discretize the parameter space $\mathcal{C}$ to get $\mathcal{C}_\epsilon$ with $|\mathcal{C}_\epsilon| \leq (3R/\epsilon)^d$ as follows.
Select any point $c \in \mathcal{C}$ and add to $\mathcal{C}_\epsilon$.
while There is a point $c' \in \mathcal{C}$ s.t. $\min_{c \in \mathcal{C}_\epsilon} ||c - c'|| > \epsilon$ do
    Add $c'$ to $c$
for $t = 1, \dots, T$ do
    Set $p_t$ as the solution of the linear system (in $N = |\mathcal{C}_\epsilon|$ variables)
        $p_{k,t} \sum_{(i,j)} \mathbf{I}[j = k] \nabla_{(i,j)} \phi(R_{(i,j),t-1}) = \sum_{l} p_{l,t} \sum_{(i,j)} \mathbf{I}[i = k] \mathbf{I}[j = l] \nabla_{(i,j)} \phi(R_{(i,j),t-1})$
    where indices $(i, j) \in \mathcal{C}_\epsilon \times \mathcal{C}_\epsilon$.
    Play according to $p_t$ and suffer loss $l_t(\cdot)$.
    Compute drift $r_{(i,j),t} = p_{j,t}(l_t(j) - l_t(i))$.
    Update $R_{(i,j),t} = R_{(i,j),t-1} + r_{(i,j),t}$.
```

By L-Lipschitzness of the loss function, if the substitution is made using some point c^* in $\mathcal{C}$ instead of points in $\mathcal{C}_\epsilon$, the learner may incur an additional regret of ϵL w.r.t. the point in $\mathcal{C}_\epsilon$ at most ϵ away from c^* (in any round t). Summing up over all rounds, we get that the internal regret of Algorithm 1 w.r.t. points in $\mathcal{C}$ is at most

$$R_T^i \leq O\left(\sqrt{dT \ln \frac{R}{\epsilon}}\right) + LT\epsilon.$$

Plugging in $\epsilon = \frac{1}{LT}$ completes the proof. □

Remark 1. *Note that for the above discretization teachnique to yield good regret it was crucial to use an algorithm on finite experts with regret sub-logarithmic in N, the number of experts. For if the regret was only say $\sqrt{N}$, or polynomial in N, (e.g. for the swap regret bound in (Blum and Mansour 2007)), substituting $\epsilon = \frac{1}{LT}$ no longer works. For an appropriate choice of ϵ their approach yields $O(T^{\frac{d+1}{d+2}} poly(d, \log T))$ regret, which has much slower convergence for any d.*

Dispersed Non-Lipschitz Loss Functions

We will now go beyond the case of Lipschitz-continuous functions. If we make no further assumption on the loss function, then linear swap regret is unavoidable. This follows from a well known 'halving' adversary which implies a $\Omega(T)$ lower bound on the external regret using piecewise constant functions. The conclusion follows by recalling that swap regret is always at least as large as the external regret. An interesting question is whether the lower bound also extends to internal regret (i.e. when only one action is swapped/rewired by the modification rules). In fact, for swap regret a general lower bound of $\Omega(\sqrt{NT\log N})$ for finite experts was recently developed by (Ito 2020). More specialized lower bounds for internal regret (which is a weaker version of swap regret) remains an open question even for the finite experts regime.

To circumvent the typical 'halving adversaries' in lower bounds, (Balcan, Dick, and Vitercik 2018) present a necessary and sufficient condition called *dispersion* for learning piecewise-Lipschitz functions. Roughly speaking, this corresponds to the assumptions that too many discontinuities (or violations of L-Lipschitzness) of the non-Lipschitz loss functions are not allowed to concentrate in any region of the action domain. Formally, this is given as

Definition 1. *Dispersion (Balcan, Dick, and Vitercik 2018). The sequence of random loss functions $l_1, \ldots, l_T$ is β-dispersed for the Lipschitz constant L if, for all T and for all $\epsilon \geq T^{-\beta}$, we have that, in expectation, at most $\tilde{O}(\epsilon T)$ functions (the soft-O notation suppresses dependence on quantities beside ϵ, T and β, as well as logarithmic terms) are not L-Lipschitz for any pair of points at distance ϵ in the domain $\mathbf{C}$. That is, for all T and for all $\epsilon \geq T^{-\beta}$,*

$$\mathbf{E}\left[\max_{\substack{\rho,\rho'\in\mathbf{C}\\ ||\rho-\rho'||_2\leq\epsilon}} \left|\{t\in[T] \mid l_t(\rho)-l_t(\rho') > L||\rho-\rho'||_2\}\right|\right] \leq \tilde{O}(\epsilon T).$$

For simplicity we will focus on piecewise constant functions in one dimension with a bounded number of discontinuities in any round. Suppose there are at most K discontinuities in any piecewise constant $l_t(\cdot)$, and the probability that a dicontinuity is located in any interval of width ϵ, $p_\epsilon \leq \kappa\epsilon$ for some $\kappa > 0$ and for any $\epsilon \geq 0$ then the above definition may be readily verified. Indeed, in any interval of width $\epsilon \geq T^{-\beta} \geq 0$, the expected number of discontinuities is $O(\kappa\epsilon)$ in any round, or $O(\kappa\epsilon T)$ across T rounds by linearity of expectation. Now by a VC-dimension based argument (see e.g. Lemma 1 of (Balcan, Dick, and Vitercik 2018)), the maximum number of discontinuities in an ϵ-interval is at most $O(\kappa\epsilon T) + O(\sqrt{T\log\frac{1}{\delta}})$ with probability at least $1-\delta$. Or the expected maximum number of discontinuities is $O\left(\kappa\epsilon T + \sqrt{T\log\frac{1}{\delta}} + \delta T\right)$. Choose $\delta = \frac{1}{T}$ to conclude that Definition 1 is satisfied for $\beta = \frac{1}{2}$. We formalize this as the following lemma.

Lemma 5. *Let $l_1, \ldots, l_T : \mathcal{C} \to [0,1]$ be a sequence of piecewise constant functions with $\mathcal{C} \subset \mathbb{R}$ and the discontinuities of the loss functions have a κ-bounded distribution*[1]*. Then the sequence of loss functions is $1/2$-dispersed in the sense of Definition 1.*

Now the key insight is that for the one-dimensional piecewise-constant case with at most K discontinuities in any function l_t, we can effectively reduce the problem to one with $N = O(KT)$ experts, one corresponding to each piece in the (also piecewise-constant) total loss function $l_T = \sum_{t=1}^{T} l_t$. The main modification needed in Algorithm 1 is in step 10, the losses used in the drift function need to scaled by the width of the constant-loss 'piece' corresponding to each expert.

Algorithm 2: CEPoM(η)

```
Input: step size $\eta \in [0,1]$, experts schedule $\mathcal{C}_t \subset 2^{\mathcal{C}}$.
Initialize $p_1$ as the uniform distribution over $\mathcal{C}_1$.
Play $a_1$ according to $p_1$.
for $t = 2, \ldots, T$ do
    Set $p_t$ as the solution of the linear system (in $N = O(KT)$ variables)
        $p_{k,t} \sum_{(i,j)} \mathbf{I}[j=k] \nabla_{(i,j)} \phi(R_{(i,j),t-1}) = \sum_{l} p_{l,t} \sum_{(i,j)} \mathbf{I}[i=k] \mathbf{I}[j=l] \nabla_{(i,j)} \phi(R_{(i,j),t-1})$
    where indices $(i,j) \in \mathcal{C}_t \times \mathcal{C}_t$.
    Play according to $p_t$ and suffer loss $l_t(\cdot)$.
    Set $r_{(i,j),t} = p_{j,t}(l_t(j)/w_t(j) - l_t(i)/w_t(j))$, where $w_t(k)$ is the width of the piece $k$ in $l_T$.
    Update $R_{(i,j),t} = R_{(i,j),t-1} + r_{(i,j),t}$.
```

Our main theorem in this section gives an upper bound on the internal regret of Continuous Exponential Potential Minimizer (CEPoM, Algorithm 2).

Theorem 6. *Consider an online learner over action space $\mathcal{C} \subset \mathbb{R}$ such that $\mathcal{C}$ is compact and closed. If the learner is faced with a sequence of L-Lipschitz losses $l_1, \ldots, l_T : \mathcal{C} \to [0,1]$ which are all piecewise-constant with at most K pieces and the discontinuities are κ-bounded, and the learner uses Algorithm 2 with $\eta = \sqrt{\frac{4}{T}\ln KT}$ and $\phi(x) = e^{\eta x}$, then there exists an experts schedule $\mathcal{C}_t$ for which the expected internal regret of the learner is $O\left(\sqrt{T\ln(KT)}\right)$.*

Remark 2. *A computationally efficient implementation in per iteration time $O(K\log(KT))$ for sampling can be achieved using the interval-tree based algorithm of (Cohen-Addad and Kanade 2017).*

Remark 3. *Extension beyond the one-dimensional piecewise constant case is possible by using techniques from (Balcan, Dick, and Vitercik 2018). Piecewise constant case is the simplest w.r.t. analysis as well as computationally efficient implementation.*

Finally, we provide a lower bound that shows our results are tight up to logarithmic factors.

Theorem 7. *Consider an online learner over compact and closed action space $\mathcal{C} \subset \mathbb{R}$. There exists a sequence of random piecewise-constant losses $l_1, \ldots, l_T : \mathcal{C} \to [0,1]$ such that any online internal regret of the learner has expected internal regret at least $\Omega\left(\sqrt{T}\right)$ on the sequence.*

[1] A distribution is said to be *κ-bounded* if the corresponding probability density $f(x)$ satisfies, $\sup_x f(x) \leq \kappa$. For example, the standard normal distribution $\mathcal{N}(\mu, \sigma)$ is $\frac{1}{\sqrt{2\pi}\sigma}$-bounded.

Bandit Feedback

For a continuous action space $\mathcal{C}$, the loss function over the entire space may be difficult to even represent, let alone compute (or observe) as a learner. Therefore it is more realistic to assume that the learner only observes partial feedback, for example only the loss value $l_t(a_t)$ corresponding to the played action $a_t \in \mathcal{C}$ instead of the entire function $l_t(\cdot)$.

The typical approach to minimize external regret in the presence of partial feedback is to estimate the losses, form a probability distribution over the experts based on these estimates, and add some uniform exploration to this distribution (in order to ensure reasonable loss estimates for different experts). Our approach to minimize the internal regret uses the same basic ingredients, but additionally adjusts its probability distribution to satisfy a fixed point constraint (similar to the linear system in the full information algorithm above).

In more detail, at each round t, the learner computes a probability distribution p_t over some subsets of the domain, given by a schedule $\mathcal{C}_t$. We use unbiased loss estimates

$$\hat{l}_t(a) = \frac{l_t(a)\mathbf{I}[a \in c_t]}{p_t(c_t)}$$

where $c_t \subseteq \mathcal{C}$ corresponds to the set from experts schedule $\mathcal{C}_t$ at round t, which was sampled according to p_t. Moreover, for each $c_i, c_j \in \mathcal{C}_t$ define the cumulative loss estimate

$$L_t^{(i,j)} = \sum_{s=1}^{t} \sum_{c \in \mathcal{C}_s} p_s^{(i,j)}(c) \int_{a \in c} \hat{l}_s(a) da$$

where $p_s^{(i,j)}(c)$ is the same as $p_s(c)$ except $p_s^{(i,j)}(c_i) = 0, p_s^{(i,j)}(c_j) = p_s(c_i) + p_s(c_j)$. For any $c_{t,k} \in \mathcal{C}_t$, we define $\omega(c_{t,k})$ as the probability mass of the uniform distribution on $\mathcal{C}$ over $c_{t,k}$ (as before). The full algorithm is presented as Algorithm 3. As before, we instantiate our algorithm for the continuous Lipschitz setting and the dispersed piecewise constant setting, and obtain respective regret bounds.

Theorem 8. *Consider an online learner over action space $\mathcal{C} \subset \mathbb{R}^d$ such that $\mathcal{C}$ is compact, closed and contained in a ball of radius R. Given a sequence of losses $l_1, \ldots, l_T : \mathcal{C} \to [0,1]$ with bandit feedback, Algorithm 3 with $\mathcal{C}_t = \mathcal{C}_\epsilon$, $\epsilon = \left(\frac{(3R)^d}{L^2T}\right)^{\frac{1}{d+2}}$, $\eta_t = \sqrt{\frac{2d}{3t}(\frac{\epsilon}{3R})^d \ln(3R/\epsilon)}$ and $\gamma_t = (\frac{3R}{\epsilon})^d \eta_t$, achieves an expected internal regret of*

$$O\left(T^{\frac{d+1}{d+2}}(\sqrt{d}R^d + L)polylog(T)\right).$$

Algorithm 3: INTERNAL EXP3(η_t, γ_t)

```
Input: step-size schedule $\eta_t \in [0,1]$, exploration schedule $\gamma_t \in [0,1]$, experts schedule $\mathcal{C}_t \subset 2^{\mathcal{C}}$.
Initialize $p_1$ as the uniform distribution over $\mathcal{C}_1$.
Play $a_1$ according to $p_1$.
for $t = 2, \ldots, T$ do
    Set $q_t^{(i,j)} = \frac{\exp(-\eta_t L_{t-1}^{(i,j)})}{\sum_{k \neq l} \exp(-\eta_t L_{t-1}^{(k,l)})}$.
    Set $\hat{p}_t$ as the solution of the linear system (in $|\mathcal{C}_t|$ variables) $\hat{p}_{t,k} = \sum_{i \neq j} q_t^{(i,j)} \hat{p}_{t,k}^{(i,j)}$, where $(i,j,k) \in \mathcal{C}_t^3$.
    Set $p_{t,k} = (1-\gamma_t)\hat{p}_{t,k} + \gamma_t \cdot \omega(c_{t,k})$.
    Play according to $p_t$ and suffer loss $l_t(\cdot)$.
```

Proof Sketch. We use discretization to cover the parameter space. In this case, the regret has unavoidable polynomial dependence on the number of experts. Therefore, we need to choose a coarser cover with $\epsilon \sim T^{\frac{-1}{d+2}}$ instead of $\epsilon \sim T^{-1}$ in order to get a sublinear regret bound. □

For the one-dimensional piecewise-constant loss setting with at most K pieces in any l_t, we use a fixed schedule $\mathcal{C}_t$ unlike the full information setting and obtain $O(T^{2/3}\text{poly}(K, \ln T))$ expected regret. The key difference is that we do not observe the actual intervals where the loss is constant. We use uniform intervals of carefully tuned width to ensure that discontinuities of the losses do not fall within our intervals with high probability, without blowing up the estimated losses $\hat{l}_t$ (which vary inversely with probability mass inside the interval).

Theorem 9. *Consider an online learner over action space $\mathcal{C} \subset \mathbb{R}$ such that $\mathcal{C}$ is compact and closed. Given a sequence of losses $l_1, \ldots, l_T : \mathcal{C} \to [0,1]$ which are all piecewise-constant with at most K pieces, there exists a parameter setting $\eta_t, \gamma_t, \mathcal{C}_t$ for Algorithm 3 such that the expected internal regret of the learner is $O\left(T^{2/3}poly(K, \ln T)\right)$.*

We conclude this section with a couple remarks.

Remark 4. *Our results can be adapted to obtain high probability bounds on the internal regret, using martingale inequalities along the lines of (Auer et al. 2002).*

Remark 5. *We have worked with the d-dimensional Euclidean space and our loss sequence may be adversarial (up to dispersion constraints). In contrast, (Kleinberg, Slivkins, and Upfal 2008) consider (external) regret bounds for a general metric space but for stochastic losses and obtain asymptotically tight results that depend on the so-called zooming dimension of the problem.*

Applications

We discuss below two main applications of our results.

Correlated Equilibria

The relationship between correlated equilibria and internal regretis well known for games with finite action spaces (Foster and Vohra 1999; Blum and Mansour 2011). Here we will define and establish the connection in the more general continuous action space setting.

Definition 2. *A game $G = \langle M, (A_i), (s_i) \rangle$ has a finite set M of m players. Player i has a continuous set $A_i \subseteq \mathcal{C}_i$ of actions and a loss function $s_i : A_i \times (\times_{j \neq i} A_j) \to [0,1]$ that maps the action of player i for any combination of actions of the other players to a bounded real number.*

The goal of each player is to minimize its loss. A *correlated equilibrium* is a distribution Q over the joint action space $A_1 \times \cdots \times A_M$ with the following property. If a vector of actions $\bar{a}$ is drawn from the distribution Q, player i is given action a_i from $\bar{a}$ (but no information regarding other players' actions). The probability distribution Q is a correlated equilibrium if, for each player i, it is their best response to play the suggested action provided that the other players do not deviate. We now formally define an ϵ-correlated equilibrium.

Definition 3. *A joint probability distribution Q over joint action space $\times A_i$ is an ϵ-correlated equilibrium w.r.t. deviations $(\mathcal{F}_k)_{k\in[M]}$ if for every player i and for any function $F : A_i \to A_i, F \in \mathcal{F}_i$, we have $\mathbf{E}_{a\sim Q}[s_i(a_i, a_{-i})] \leq \mathbf{E}_{a\sim Q}[s_i(F(a_i), a_{-i})] + \epsilon$, where a_{-i} denotes the joint actions of the other players besides player i.*

In other words, P is an ϵ-correlated equilibrium if the expected incentive to deviate is at most ϵ for every player. By choosing the class $\mathcal{F}_k$ to one that allows swaps of single actions in A_k, we obtain a direct connection with internal regret. The following theorem relates the empirical distribution of the actions performed by each player, their swap regret, and the distance from a correlated equilibrium (generalizes finite action space version of Foster and Vohra 1998; Hart and Mas-Colell 2000).

Theorem 10. *Let $G = \langle M, (A_i), (s_i)\rangle$ be a game and assume that for T time steps each player follows a strategy that has internal regret of at most R_T. The empirical distribution $\hat{Q}$ of the joint actions played by the players is an (R_T/T)-correlated equilibrium, and the loss of each player equals its expected loss on $\hat{Q}$.*

The above theorem states that the payoff for each player is their payoff in some approximate correlated equilibrium. In addition, the theorem relates the internal regret to the distance from a correlated equilibrium. Note that if the average internal regret vanishes then the game converges in the limit to the set of correlated equilibria. Our internal regret algorithms therefore give strategies to achieve correlated equilibrium in a game with L-Lipschitz (Azrieli and Shmaya 2013; Deligkas, Fearnley, and Spirakis 2020) or piecewise-constant losses (for example item-pricing and auction design Morgenstern and Roughgarden 2016; Syrgkanis 2017; Balcan, Sandholm, and Vitercik 2018).

Data-driven Hyperparameter Tuning

Data-driven algorithm selection or parameter tuning is an approach to tune continuous hyperparameters of an algorithm, by learning over multiple input instances of the problem (Gupta and Roughgarden 2016). The approach has found impactful application to several fundamental problems, including semi-supervised learning, clustering, linear regression, adversarial robustness and simulated annealing (Balcan and Sharma 2021; Balcan et al. 2017, 2022, 2023; Balcan, Nguyen, and Sharma 2023; Blum, Dan, and Seddighin 2021). The loss-functions (as a function of the parameter) for combinatorial optimization algorithms like clustering and greedy knapsack are typically piecewise constant as algorithms make different (discrete) choices across the breakpoints (Balcan 2020). Prior work has shown bounded external and tracking regret for online hyperparameter tuning for these algorithms, also assuming the dispersion condition is satisfied (Balcan, Dick, and Vitercik 2018; Sharma, Balcan, and Dick 2020). Our work achieves vanishing internal regret for this problem under the same assumptions.

Formally, for a given algorithmic problem (say clustering, or knapsack), let Π denote the set of problem instances of interest. We also fix a (potentially infinite) family of algorithms $\mathcal{A}$, parameterized by a set $\mathcal{P} \subseteq \mathbb{R}^d$. Let A_ρ denote the algorithm in the family $\mathcal{A}$ parameterized by $\rho \in \mathcal{P}$. The performance of any algorithm on any problem instance is given by a utility function $u : \Pi \times \mathcal{P} \to [0, H]$, i.e. $u(x, \rho)$ measures the performance on problem instance $x \in \Pi$ of algorithm $A_\rho \in \mathcal{A}$. The utility of a fixed algorithm A_ρ from the family is given by $u_\rho : \Pi \to [0, H]$, with $u_\rho(x) = u(x, \rho)$. In our online learning setup, the learner receives the dual class function $u_x : \mathcal{P} \to [0, H]$, with $u_x(\rho) = u_\rho(x)$, which measure the performance of all algorithms of the family for a fixed problem instance $x \in \Pi$ (in particular, the dual function for problem instance x_t in round t). In the bandit feedback setting, one only receives $u_x(\rho_t)$ for the parameter ρ_t played by the learner. For many parameterized algorithms, the dual class functions are piecewise constant (Balcan 2020). We instantiate our results for the knapsack problem.

Greedy Knapsack. Knapsack is a well-known NP-complete problem. We are given a knapsack with capacity $\texttt{cap}$ and items $i \in [m]$ with sizes w_i and values v_i. The goal is to select a subset S of items to add to the knapsack such that $\sum_{i\in S} w_i \leq \texttt{cap}$ while maximizing the total value $\sum_{i\in S} v_i$ of selected items. The greedy heuristic to add items in decreasing order of v_i/w_i gives a 2-approximation. We consider a generalization to use v_i/w_i^ρ proposed by (Gupta and Roughgarden 2016) for $\rho \in [0, 10]$. For example, for the value-weight pairs $\{(0.99, 1), (0.99, 1), (1.01, 1.01)\}$ and capacity $\texttt{cap} = 2$ the classic heuristic $\rho = 1$ gives value 1.01 but using $\rho = 3$ gives the optimal value 1.98. We can tune the parameter ρ with vanishing internal regret.

Theorem 11. *Consider instances of the knapsack problem given by bounded weights $w_t \in [1, C]$ and independent values $v_t \in [0, 1]$ drawn from some bounded-density distribution (which may change with t) for $t \in [T]$. Then there is an algorithm for learning the parameter ρ for the greedy heuristic family above with expected internal regret $\tilde{O}(\sqrt{T})$.*

Conclusion

We provide a novel extension of the notion of internal regret on continuous action spaces, and algorithms that achieve no regret guarantees in full and bandit information settings. Internal regret based strategies lead to correlated equilibria, which we expect to be impactful in automated mechanism design, and data-driven algorithm design more broadly. Our research raises several interesting questions for future research including extension to other notions of regret (e.g. swap/transductive/dynamic regret), lower bounds in the bandit setting and computational efficiency for large action space dimension d.

Acknowledgments

The direction was inspired by a comment from Nina Balcan during a guest lecture by Avrim Blum. We appreciate feedback from Keegan Harris and anonymous reviewers.

References

Auer, P.; Cesa-Bianchi, N.; Freund, Y.; and Schapire, R. E. 2002. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1): 48–77.

Aumann, R. J. 1974. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1): 67–96.

Azrieli, Y.; and Shmaya, E. 2013. Lipschitz games. *Mathematics of Operations Research*, 38(2): 350–357.

Balcan, M.-F. 2020. Data-Driven Algorithm Design. In Roughgarden, T., ed., *Beyond Worst Case Analysis of Algorithms*. Cambridge University Press.

Balcan, M.-F.; Blum, A.; Sharma, D.; and Zhang, H. 2023. An analysis of robustness of non-lipschitz networks. *Journal of Machine Learning Research (JMLR)*, 24(98): 1–43.

Balcan, M.-F.; Dick, T.; and Vitercik, E. 2018. Dispersion for data-driven algorithm design, online learning, and private optimization. In *Symposium on Foundations of Computer Science (FOCS)*, 603–614. IEEE.

Balcan, M.-F.; Khodak, M.; Sharma, D.; and Talwalkar, A. 2022. Provably tuning the ElasticNet across instances. *Advances in Neural Information Processing Systems (NeurIPS)*, 35: 27769–27782.

Balcan, M.-F.; Nagarajan, V.; Vitercik, E.; and White, C. 2017. Learning-theoretic foundations of algorithm configuration for combinatorial partitioning problems. In *Conference on Learning Theory (COLT)*, 213–274. PMLR.

Balcan, M.-F.; Sandholm, T.; and Vitercik, E. 2018. A general theory of sample complexity for multi-item profit maximization. In *Conference on Economics and Computation (EC)*, 173–174.

Balcan, M.-F.; and Sharma, D. 2021. Data driven semi-supervised learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 34: 14782–14794.

Balcan, N.; Nguyen, A. T.; and Sharma, D. 2023. New Bounds for Hyperparameter Tuning of Regression Problems Across Instances. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Bartlett, P.; Indyk, P.; and Wagner, T. 2022. Generalization bounds for data-driven numerical linear algebra. In *Conference on Learning Theory (COLT)*, 2013–2040. PMLR.

Blackwell, D. 1956. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1): 1–8.

Blum, A. 1997. Empirical support for winnow and weighted-majority algorithms: Results on a calendar scheduling domain. *Machine Learning*, 26(1): 5–23.

Blum, A.; Dan, C.; and Seddighin, S. 2021. Learning complexity of simulated annealing. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 1540–1548. PMLR.

Blum, A.; and Lykouris, T. 2020. Advancing Subgroup Fairness via Sleeping Experts. In *Innovations in Theoretical Computer Science Conference (ITCS)*, volume 11.

Blum, A.; and Mansour, Y. 2007. From external to internal regret. *Journal of Machine Learning Research (JMLR)*, 8(6).

Blum, A.; and Mansour, Y. 2011. Learning, regret minimization, and equilibria. *Algorithmic Game Theory (Chapter 4)*.

Cesa-Bianchi, N.; and Lugosi, G. 2003. Potential-based algorithms in on-line prediction and game theory. *Machine Learning*, 51(3): 239–261.

Cohen, W. W.; and Singer, Y. 1996. Learning to query the web. In *AAAI Workshop on Internet-Based Information Systems*, 16–25.

Cohen, W. W.; and Singer, Y. 1999. Context-sensitive learning methods for text categorization. *ACM Transactions on Information Systems (TOIS)*, 17(2): 141–173.

Cohen-Addad, V.; and Kanade, V. 2017. Online optimization of smoothed piecewise constant functions. In *Artificial Intelligence and Statistics (AISTATS)*, 412–420. PMLR.

Deligkas, A.; Fearnley, J.; and Spirakis, P. 2020. Lipschitz continuity and approximate equilibria. *Algorithmica*, 82(10): 2927–2954.

Foster, D. P.; and Vohra, R. 1999. Regret in the on-line decision problem. *Games and Economic Behavior*, 29(1-2): 7–35.

Foster, D. P.; and Vohra, R. V. 1998. Asymptotic calibration. *Biometrika*, 85(2): 379–390.

Freund, Y.; Schapire, R. E.; Singer, Y.; and Warmuth, M. K. 1997. Using and combining predictors that specialize. In *Symposium on Theory of Computing (STOC)*, 334–343.

Gupta, R.; and Roughgarden, T. 2016. A PAC approach to application-specific algorithm selection. In *Innovations in Theoretical Computer Science (ITCS)*, 123–134.

Hart, S.; and Mas-Colell, A. 2000. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5): 1127–1150.

Ito, S. 2020. A tight lower bound and efficient reduction for swap regret. *Advances in Neural Information Processing Systems (NeurIPS)*, 33: 18550–18559.

Kleinberg, R.; Niculescu-Mizil, A.; and Sharma, Y. 2010. Regret bounds for sleeping experts and bandits. *Machine learning*, 80(2): 245–272.

Kleinberg, R.; Slivkins, A.; and Upfal, E. 2008. Multi-armed bandits in metric spaces. In *ACM Symposium on Theory of Computing (STOC)*, 681–690.

Lehrer, E. 2003. A wide range no-regret theorem. *Games and Economic Behavior*, 42(1): 101–115.

Mohri, M.; and Yang, S. 2014. Conditional swap regret and conditional correlated equilibrium. *Advances in Neural Information Processing Systems (NeurIPS)*, 27.

Mohri, M.; and Yang, S. 2017. Online learning with transductive regret. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

Morgenstern, J.; and Roughgarden, T. 2016. Learning simple auctions. In *Conference on Learning Theory (COLT)*, 1298–1318. PMLR.

Seneta, E. 1980. Computing the stationary distribution for infinite Markov chains. *Linear Algebra and Its Applications*, 34: 259–267.

Sharma, D.; Balcan, M.-F.; and Dick, T. 2020. Learning piecewise Lipschitz functions in changing environments. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 3567–3577. PMLR.

Stoltz, G. 2005. *Incomplete information and internal regret in prediction of individual sequences*. Ph.D. thesis, Université Paris Sud-Paris XI.

Stoltz, G.; and Lugosi, G. 2005. Internal regret in on-line portfolio selection. *Machine Learning*, 59(1): 125–159.

Stoltz, G.; and Lugosi, G. 2007. Learning correlated equilibria in games with compact sets of strategies. *Games and Economic Behavior*, 59(1): 187–208.

Syrgkanis, V. 2017. A sample complexity measure with applications to learning optimal auctions. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.