Learning with Stochastic Inputs and Adversarial Outputs

Alessandro Lazaric, Rémi Munos

SequeL Project, INRIA Lille - Nord Europe, France

Abstract

Most of the research in online learning is focused either on the problem of adversarial classification (i.e., both inputs and labels are arbitrarily chosen by an adversary) or on the traditional supervised learning problem in which samples are independent and identically distributed according to a stationary probability distribution. Nonetheless, in a number of domains the relationship between inputs and outputs may be adversarial, whereas input instances are i.i.d. from a stationary distribution (e.g., user preferences). This scenario can be formalized as a learning problem with stochastic inputs and adversarial outputs. In this paper, we introduce this novel stochastic-adversarial learning setting and we analyze its learnability. In particular, we show that in binary classification, given a hypothesis space $\mathcal{H}$ with finite VC-dimension, it is possible to design an algorithm which incrementally builds a suitable finite set of hypotheses from $\mathcal{H}$ used as input for an exponentially weighted forecaster and achieves a cumulative regret of order $O(\sqrt{nVC(\mathcal{H})\log n})$ with overwhelming probability. This result shows that whenever inputs are i.i.d. it is possible to solve any binary classification problem using a finite VC-dimension hypothesis space with a sub-linear regret independently from the way labels are generated (either stochastic or adversarial). We also discuss extensions to multi-label classification, regression, learning from experts and bandit settings with stochastic side information, and application to games.

Key words: On-line Learning, Hybrid Stochastic-Adversarial Learning

Email addresses: `alessandro.lazaric@inria.fr` (Alessandro Lazaric), `remi.munos@inria.fr` (Rémi Munos)

URL: `http://sequel.futurs.inria.fr/lazaric` (Alessandro Lazaric), `http://sequel.futurs.inria.fr/munos` (Rémi Munos)

1. Introduction

Motivation and relevance. The problem of classification has been intensively studied both in the *stochastic* and *adversarial* settings. In the former, inputs and labels are jointly drawn from a stationary probability distribution, while in the latter no assumption is made on the way the sequence of input-label pairs is generated. Although the adversarial setting allows to consider a wide range of problems by dropping any assumption about the way data are generated, in many applications it is possible to consider an hybrid scenario in which inputs are independent and identically distribution (i.i.d.) from a distribution and labels are adversarially chosen. Let us consider a classification problem in which a company tries to predict whether a user is likely to buy an item or not (e.g., a new model of mobile phone, a new service) on the basis of a set of features describing her profile (e.g., sex, age, salary, etc.). In the medium-term, user profiles can be well assumed as coming from a stationary distribution. In fact, features such as age and salary are almost constant and their distribution in a sample set does not change in time. On the other hand, user preferences may rapidly change in an unpredictable way (e.g., because of competitors who released a new product). This scenario can be formalized as a classification problem with stochastic inputs and adversarial labels. Alternatively, the problem can be casted as a two-player games in which the structure of the game (i.e., the payoffs) is determined by a stochastic event x (e.g., a card, a dice). At each round both the learner and the adversary select a strategy h defined over all the possible events and plays the corresponding action $h(x)$. In general, the resulting payoff for the two players is a function of the actions and the stochastic event x. The Nash equilibrium of such a game is a pair of mixed strategies (i.e., a probability distribution over the set of pure strategies) such that their expected payoff (where expectation is taken on strategies randomization and the event distribution) cannot be improved by unilateral deviations from equilibrium strategies.

Definition of the general problem. More formally, we consider the general prediction problem summarized in the *protocol* in Figure 1. At each round t an input x_t is drawn from a stationary distribution P (unknown to the learner) and revealed to both the learner and the adversary. Simultaneously, the adversary chooses a loss function ℓ_t and the learner chooses a

for $t = 1, 2, \ldots$ **do**
  A sample $x_t \overset{iid}{\sim} P$ is revealed to both the learner and the adversary
  Simultaneously,
    - Adversary chooses a loss function $\ell_t : \mathcal{Y} \to [0, 1]$
    - Learner chooses a hypothesis $h_t \in \mathcal{H}$
  Learner predicts $\widehat{y}_t = h_t(x_t) \in \mathcal{Y}$
  Learner observes the feedback:
    - $\ell_t(\widehat{y}_t)$ (in case of *bandit* information)
    - *or* $\ell_t(\cdot)$ (in case of *full* information)
  Learner incurs a loss $\ell_t(\widehat{y}_t)$
end for

Figure 1: The protocol of the general stochastic-adversarial setting.

hypothesis h_t in a hypothesis space $\mathcal{H}$ and predicts $\widehat{y}_t = h_t(x_t)$. The feedback returned to the learner can be either the loss function ℓ_t (i.e., *full* information) or just the loss $\ell_t(\widehat{y}_t)$ of the chosen prediction (i.e., *bandit* information). The objective of the learner is to minimize her regret, that is to incur a cumulative loss that is almost as small as the one obtained by the best hypothesis in $\mathcal{H}$ on the same sequence of inputs drawn from P and loss functions provided by the adversary. Formally, for any $n > 0$, the regret of an algorithm $\mathcal{A}$ is

$$R_n(\mathcal{A}) = \sum_{t=1}^{n} \ell_t(h_t(x_t)) - \inf_{h \in \mathcal{H}} \sum_{t=1}^{n} \ell_t(h(x_t)), \tag{1}$$

where h_t is the hypothesis chosen by $\mathcal{A}$ at time t.

Results so far. In the full information adversarial setting, many theoretical results are available for online learning algorithms with different hypothesis spaces.

Finite spaces. Given a finite set of N experts (i.e., hypotheses) as input, at each round the exponentially weighted forecaster (EWF) (Littlestone & Warmuth, 1994; Cesa-Bianchi et al., 1997; Vovk, 1998) randomizes on experts' predictions with a probability concentrated on experts which had a good performance so far (i.e., low cumulative loss). Despite its simplicity,

the EWF achieves a regret upper-bounded by $O(\sqrt{n \log N})$, where n is the time horizon of the problem. Although the logarithmic dependency on the number of experts allows the use of a large number of experts, the EWF cannot be directly extended to the case of infinite sets of experts.

Linear spaces. Many margin based algorithms with linear hypotheses have been proposed for adversarial classification (Rosenblatt, 1958; Weston & Watkins, 1999; Crammer & Singer, 2003). The simplest example of this class of algorithms is the perceptron (Rosenblatt, 1958) in which a weight matrix W is updated whenever a prediction mistake is made. The number of classification mistakes of the perceptron is bounded (Fink et al., 2006) by $L + D + \sqrt{LD}$ where L is the cumulative loss and D is the complexity of any weight matrix. In the linearly separable case (i.e., input-label pairs can be perfectly classified by a linear classifier, that is $L = 0$), the number of mistake is finite (for any time horizon n) and depends on the complexity D of the weight matrix corresponding to the optimal classifier.

General spaces. The agnostic online learning algorithm recently proposed by Ben-David et al. (2009) successfully merges the effectiveness of the EWF with the general case of an infinite hypothesis set $\mathcal{H}$. Under the assumption that the Littlestone dimension (Littlestone, 1988) of $\mathcal{H}$ is finite ($Ldim(\mathcal{H}) < \infty$), it is possible to define a suitable finite subset of $\mathcal{H}$ such that the EWF achieves a regret of the order of $O(Ldim(\mathcal{H}) + \sqrt{nLdim(\mathcal{H}) \log n})$.

The problem of classification with partial information (also known as contextual bandit problem) is of major interest in applications in which the true label is not revealed and only the loss for the chosen label is returned to the learner (e.g., recommendation systems). This scenario is analyzed by Langford & Zhang (2007) in the fully stochastic setting. They introduce an epoch-based online learning algorithm whose regret can be bounded by merging supervised sample bounds with bandit bounds. Kakade et al. (2008) propose a modification of the perceptron (i.e., the *banditron*) to solve the online multi-label classification problem in the fully adversarial case. In particular, they analyze the performance of the banditron in terms of mistake bounds with particular attention to the linearly separable case.

Contributions. While all the previous approaches consider either the fully adversarial or fully stochastic setting, in this paper, we analyze the problem of prediction in case of stochastic inputs and adversarial loss functions. The main contributions of this paper can be summarized as follows: *(i)* introduction of the stochastic-adversarial learning setting, *(ii)* design of an online learning algorithm with polynomial complexity in n (with exponent the VC-

dimension of $\mathcal{H}$) achieving a sub-linear regret, *(iii)* analysis of the learnability of the stochastic-adversarial setting revealing the same complexity measure as the fully stochastic setting, *(iv)* extension to other learning scenarios such as partial information, regression, and games.

Outline. In Section 2, we consider a specific instance of the general problem of Figure 1, that is the problem of binary classification with full information. In Section 3 we drop any assumption about the distribution and the existence of auxiliary samples and devise an epoch-based algorithm that, given a hypothesis set $\mathcal{H}$ as input, incrementally builds a finite subset of $\mathcal{H}$ on the basis of the sequence of inputs experienced so far. At the beginning of each epoch, a new subset of $\mathcal{H}$ is generated and it is given as input to a EWF which is run until the end of the epoch. Because of the stochastic assumption about the generation of inputs, the complexity of the hypothesis space $\mathcal{H}$ can be measured according to the VC-dimension instead of the Littlestone dimension like in the agnostic online learning algorithm. As a result, the algorithm performance can be directly obtained by merging the EWF performance in the adversarial setting and usual capacity measures for hypothesis spaces in stochastic problems (e.g., their VC-dimension). The resulting algorithm is proved to incur a regret of order $O(\sqrt{nVC(\mathcal{H})\log n})$ with overwhelming probability. The computational complexity of the algorithm is discussed in Section 4. A number of extensions are then considered in Section 5 for multi-label prediction, regression, bandit information, and games with stochastic side information. Section 6 compares the proposed algorithm with existing online learning algorithms for the stochastic or adversarial setting. Finally, in Section 7 we draw conclusions.

2. The Problem

Notation. In this section, we formally define the problem of binary classification and we introduce the notation used in the rest of the paper. Let $\mathcal{X}$ be the input space, P a probability distribution defined on $\mathcal{X}$, and $\mathcal{Y} = \{0, 1\}$ the set of labels. The learner is given as input a (possibly infinite) set $\mathcal{H}$ of hypotheses of the form $h : \mathcal{X} \to \mathcal{Y}$, mapping each possible input to a label. We define the *disagreement* between two hypotheses $h, h' \in \mathcal{H}$ as

$$\Delta(h, h') = \mathbb{E}_{x\sim P}\left[\mathbb{I}\left\{h(x) \neq h'(x)\right\}\right], \tag{2}$$

(where $\mathbb{I}\{\xi\} = 1$ when event ξ is true, and 0 otherwise) that is, the probability that h and h' make different predictions given inputs drawn from P.

1: **for** $t = 1, 2, \ldots$ **do**
2: A sample $x_t \overset{iid}{\sim} P$ is revealed to both the learner and the adversary
3: Simultaneously,

- Adversary chooses a label $y_t \in \mathcal{Y} = \{0, 1\}$
- Learner chooses a hypothesis $h_t \in \mathcal{H}$

4: Learner predicts $\widehat{y}_t = h_t(x_t) \in \mathcal{Y}$
5: y_t is revealed
6: Learner incurs a loss $\ell(\widehat{y}_t, y_t) = \mathbb{I}\{\widehat{y}_t \neq y_t\}$
7: **end for**

Figure 2: The protocol of the fully information binary stochastic-adversarial classification problem.

The protocol. The on-line classification problem we consider is summarized in Figure 2. The main difference with the general setting (Figure 1) is that at each round t the adversary chooses a label y_t [1], and the learner incurs a loss $\ell(\widehat{y}_t, y_t)$ defined as $\mathbb{I}\{\widehat{y}_t \neq y_t\}$. In the following, we will use the short form $\ell_t(h)$ for $\ell(h(x_t), y_t)$ with $h \in \mathcal{H}$. Since at the end of each round the true label y_t is explicitly revealed (i.e., *full information* feedback), the learner can compute the loss for any hypothesis in $\mathcal{H}$. The objective of the learner is to minimize regret (1). As it can be noticed, in general the loss $\ell_t(h_t)$ is a random variable that depends on the (random) loss function ℓ_t chosen by the adversary, the (randomized) algorithm, and the distribution P. In the following, we consider the case of non-oblivious adversaries, so that the sequence of functions ℓ_t is fixed in advance. Thus, all the results presented in the paper will be stated in high-probability with respect to two sources of randomness: the algorithm and the samples. In the next section, we introduce the Epoch-based Stochastic Adversarial (EStochAd) forecaster for the classification problem with stochastic inputs and adversarial labels.

[1] In the general case of a non-oblivious adversary, y_t may depend on past inputs $\{x_s\}_{s<t}$, predictions $\{\widehat{y}_s\}_{s<t}$, and current input x_t.

3. Hybrid Stochastic-Adversarial Algorithms

3.1. Finite hypothesis space

Before entering in details about the algorithm, we briefly recall the EWF with side information with a finite number of experts. Let the hypothesis space $\mathcal{H}$ contain $N < \infty$ hypotheses (i.e., experts). At time t, each hypothesis h_i ($i \in \{1, \ldots, N\}$) has a weight

$$w_i^t = \exp\left(-\eta \sum_{s=1}^{t-1} \ell_s(h_i)\right) \tag{3}$$

where η is a strictly positive parameter. According to the previous definition, the smaller the cumulative loss the higher the weight for the hypothesis. At each step t, a loss function ℓ_t is adversarially chosen and at the same time, the EWF randomly selects a hypothesis h_t according to a distribution $\mathbf{p}^t = (p_1^t, \ldots, p_N^t)$, where $p_i^t = \frac{w_i^t}{\sum_{j=1}^N w_j^t}$. As a result, it incurs a loss $\ell_t(h_t)$. At the end of each round, weights are recomputed according to (3) (or updated using an incremental version of (3)). The following result provides an upper-bound on the regret for EWF .

Theorem 1. ***(see Cesa-Bianchi & Lugosi, 2006, pg. 72)*** *Let $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, 1]$ be a loss function. For any $n, N \geq 1$, $0 \leq \beta \leq 1$, $\eta > 0$ and $w_i^1 = 1$, $i \in \{1, \ldots, N\}$ the exponentially weighted average forecaster satisfies*

$$\begin{aligned} R_n &= \sum_{t=1}^n \ell_t(h_t) - \min_{h \in \mathcal{H}} \sum_{t=1}^n \ell_t(h) \\ &\leq \frac{\log N}{\eta} + \frac{n\eta}{8} + \sqrt{\frac{n}{2} \log \frac{1}{\beta}}, \end{aligned}$$

with probability at least $1 - \beta$. Optimizing the parameter $\eta = \sqrt{8 \log N / n}$, the bound becomes

$$R_n \leq \sqrt{\frac{n \log N}{2}} + \sqrt{\frac{n}{2} \log \frac{1}{\beta}}. \tag{4}$$

The implicit assumption in the previous theorem is that the time horizon n is known in advance. As usual, it is possible to obtain an anytime result for the previous algorithm by setting the learning parameter η to be a decreasing

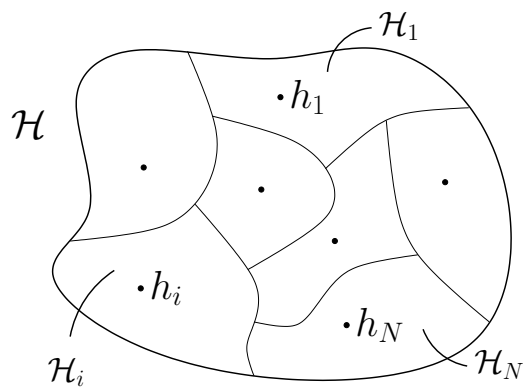

Figure 3: The hypothesis space $\mathcal{H}$ can be partitioned into classes containing hypotheses with the same sequence of prediction on inputs $\{x_t\}_{t=1}^n$. The grid H_n is obtained by selecting one hypothesis for each class of the partition.

function of t (see e.g. Auer et al. (2003)). As it can be noticed, the EWF has a logarithmic dependency on the number of experts, thus allowing to consider large sets of experts. Nonetheless, the EWF cannot be directly applied when $\mathcal{H}$ contains an infinite number of hypotheses. In next sections we show that when inputs are drawn from a fixed distribution and the hypothesis space has a finite VC-dimension, it is possible to incrementally define a finite subset of $\mathcal{H}$ that can be used as input for a EWF with a regret of the same order as in (4).

3.2. Infinite hypothesis space

Sequence of inputs known in advance. First we show that for any finite VC-dimension hypothesis space $\mathcal{H}$ and any sequence of inputs, it is possible to define in *hindsight* a finite subset $H \subset \mathcal{H}$ that contains hypotheses with exactly the same performance as those in the full set $\mathcal{H}$. Let $VC(\mathcal{H}) = d < \infty$ and $\{x_t\}_{t=1}^n$ be a sequence of i.i.d. inputs drawn from P. On the basis of $\{x_t\}_{t=1}^n$, we define a partition $\mathcal{P}_n = \{\mathcal{H}_i\}_{i \leq N}$ of $\mathcal{H}$, such that each class $\mathcal{H}_i$ contains hypotheses with the same sequence of predictions up to time n (i.e., $\forall h, h' \in \mathcal{H}_i, h(x_s) = h'(x_s), \forall s \leq n$). From each class we pick an arbitrary hypothesis $h_i \in \mathcal{H}_i$ and we define the grid $H_n = \{h_i\}_{i \leq N}$. Since $\mathcal{H}$ has a finite VC-dimension, for any $n > 0$ the cardinality of H_n is bounded by $N = |H_n| \leq \left(\frac{en}{d}\right)^d < \infty$ (Sauer, 1972). The grid H_n can also be incrementally refined as inputs are revealed. For instance, after observing x_1, $\mathcal{H}$ is partitioned in two classes containing hypotheses which predict 0 in x_1 and those which

predict 1 respectively. The set H_1 is obtained by choosing arbitrarily any two hypotheses from the two classes. As new inputs are observed each class may be further split (see Figure 3) and after n inputs the hypothesis space is partitioned into at most $O(n^d)$ classes. Finally, H_n is obtained by taking one hypothesis from each class. As a result, for any hypothesis in $\mathcal{H}$ there exists a corresponding hypothesis in H_n which has exactly the same sequence of predictions on $\{x_t\}_{t=1}^n$ and, thus, the very same performance.

Lemma 1. *Let H_n be the grid defined above, then*

$$\inf_{h\in\mathcal{H}} \sum_{t=1}^n \ell_t(h) = \min_{h'\in H_n} \sum_{t=1}^n \ell_t(h'), \tag{5}$$

that is, the performance of the best hypothesis in $\mathcal{H}$ on $\{x_t\}_{t=1}^n$ is exactly the same obtained by the best hypothesis in H_n.

Proof. The statement follows by construction. In fact, by definition of H_n, for any $h \in \mathcal{H}$ it is always possible to find a hypothesis $h' \in H_n$ with exactly the same sequence of predictions on inputs $\{x_t\}_{t=1}^n$. □

According to the previous lemma, if the sequence of inputs is available before the learning to take place, then the regret defined in (1) (that compares the cumulative loss of the learner to the performance of the best hypothesis in the full set $\mathcal{H}$) can be controlled by a EWF run on H_n, thus obtaining exactly the same performance as in Theorem 1.

Lemma 2. *Let the sequence of inputs $x_1, \ldots, x_n \overset{iid}{\sim} P$ be available before learning and let H_n be the grid defined above, then*

$$R_n \leq \sqrt{\frac{nd}{2}\log\frac{en}{d}} + \sqrt{\frac{n}{2}\log\frac{1}{\beta}}$$

with probability $1-\beta$.

Proof. The lemma immediately follows from Lemma 1, Theorem 1, and $N \leq \left(\frac{en}{d}\right)^d$ from Sauer (1972)'s lemma. □

It is interesting to notice that a similar result is derived in Kakade & Kalai (2005) for online transductive learning in which no assumption is made on the way inputs $\{x_t\}_{t=1}^n$ are generated. Thus, the performance in the bound of

Lemma 2 can be attained with both stochastic or adversarial inputs. As we show next this will be no longer the case when we move from the transductive (e.g., inputs known in advance) to the general setting (e.g., both inputs and labels are revealed online).

Sequence of auxiliary inputs. In our case, the sequence of inputs $\{x_t\}_{t=1}^n$ is not available beforehand, thus it is not possible to build H_n before the actual learning process begins. Nonetheless, in the following we show that in case of stochastic inputs, the learner can take advantage of any sequence of inputs drawn from the same distribution P to build a grid H that can be used as input for a EWF . We will further show in Section 3.3 that we do not even need to know a sequence of auxiliary inputs beforehand and the mere assumption that inputs are drawn from a fixed (and unknown) distribution is sufficient to learn efficiently.

For now, let us assume an auxiliary sequence of n' inputs $\{x'_t\}_{t=1}^{n'}$ is available to the learner before the classification problem actually begins and let $H_{n'}$ be the grid of $\mathcal{H}$ built on inputs $\{x'_t\}_{t=1}^{n'}$. The regret of EWF with experts in $H_{n'}$ can be decomposed as

$$\begin{aligned} R_n &= \sum_{t=1}^n \ell_t(h_t) - \inf_{h\in\mathcal{H}} \sum_{t=1}^n \ell_t(h) \\ &= \left(\sum_{t=1}^n \ell_t(h_t) - \min_{h'\in H_{n'}} \sum_{t=1}^n \ell_t(h')\right) + \left(\min_{h'\in H_{n'}} \sum_{t=1}^n \ell_t(h') - \inf_{h\in\mathcal{H}} \sum_{t=1}^n \ell_t(h)\right) \\ &= R_{EWF} + R_H, \end{aligned} \tag{6}$$

where R_{EWF} is the regret due to EWF and R_H comes from the use of $H_{n'}$ instead of the full hypothesis space $\mathcal{H}$. While the first term can be bounded as in Theorem 1, the second term in general is strictly positive. In fact, since $H_{n'}$ is different from the set H_n that would be created according to the inputs $\{x_t\}_{t=1}^n$, equality (5) does not hold for $H_{n'}$. In particular, in the fully adversarial case, the sequence of inputs could be chosen so that hypotheses in $H_{n'}$ have an arbitrarily bad performance when used to learn on $\{x_t\}_{t=1}^n$ (e.g., if the learner is shown the same input for n' steps, $H_{n'}$ would contain only two hypotheses!). The situation is different in the stochastic-adversarial setting. In fact, since all the inputs are sampled from the same distribution P, $H_{n'}$ is likely to contain hypotheses that are good to predict on any other sequence of inputs drawn from P. Therefore, under the assumption that n' inputs can be sampled from P beforehand, we prove that the regret (6) is bounded by $O(\sqrt{nd\log n'})$ with high probability.

Let

$$\Delta_n(h,h') = \frac{1}{n}\sum_{t=1}^{n} \mathbb{I}\left\{h(x_t) \neq h'(x_t)\right\} \tag{7}$$

be the empirical disagreement between two hypotheses $h, h' \in \mathcal{H}$ on a sequence of inputs $\{x_t\}_{t=1}^n$ (and define similarly $\Delta_{n'}(h,h')$ as the empirical disagreement of h and h' on inputs $\{x'_t\}_{t=1}^{n'}$). The following result states the uniform concentration property of Δ_n around its expectation Δ.

Lemma 3. *For any sequence of inputs $x_1, \ldots, x_n \overset{iid}{\sim} P$*

$$\sup_{(h,h')\in\mathcal{H}^2} |\Delta_n(h,h') - \Delta(h,h')| \leq \varepsilon_n = 2\sqrt{2\frac{2d\log\frac{en}{d} + \log\frac{4}{\beta}}{n}},$$

with probability $1-\beta$.

Proof. Let $\mathcal{G} = \{g(x) = \mathbb{I}\left\{h(x) \neq h'(x)\right\}, h, h' \in \mathcal{H}\}$, with $g \in \{0,1\}$. As it can be noticed $\Delta_n(h,h')$ and $\Delta(h,h')$ are respectively the empirical average and expectation of g. Furthermore, it easy to show that $VC(\mathcal{G}) = VC(\mathcal{H}^2) \leq 2VC(\mathcal{H}) = 2d$. Using the VC-bound on space $\mathcal{G}$ (see e.g., Bousquet et al. (2004)) the statement follows. □

Using the previous lemma, it is possible to bound the difference in performance between the best hypothesis in $H_{n'}$ and the best in $\mathcal{H}$, and bound the regret in (6).

Theorem 2. *For any $0 < n \leq n'$, let $H_{n'}$ be a set of hypoheses built according to an auxiliary sequence of inputs $x'_1, \ldots, x'_{n'} \overset{iid}{\sim} P$. An EWF with experts in $H_{n'}$ run on n new samples drawn from distribution P incurs a regret*

$$R_n \leq c_1\sqrt{\frac{nd}{2}\log\frac{en'}{d}} + c_2\sqrt{\frac{n}{2}\log\frac{12}{\beta}} \tag{8}$$

with probability $1-\beta$, where $c_1 = 1 + 8\sqrt{2}$, $c_2 = 9$.

Proof. In (6) the regret is decomposed in two terms. By bounding the first term as in Theorem 1, we obtain

$$\begin{aligned} R_n &\leq \sqrt{\frac{nd}{2}\log\frac{en'}{d}} + \sqrt{\frac{n}{2}\log\frac{1}{\beta'}} \\ &+ \left(\min_{h'\in H_{n'}}\sum_{t=1}^{n}\ell_t(h') - \inf_{h\in\mathcal{H}}\sum_{t=1}^{n}\ell_t(h)\right), \end{aligned}$$

with probability $1-\beta'$, where the number of hypotheses in $H_{n'}$ is bounded by $|H_{n'}| \leq (en'/d)^d$. Since both $(x'_1, \dots, x'_{n'})$ and $(x_1, \dots, x_n)$ are drawn from the same distribution, the second term can be bounded as follows

$$
\begin{aligned}
R_H &= \left(\min_{h' \in H_{n'}} \sum_{t=1}^{n} \ell_t(h') - \inf_{h \in \mathcal{H}} \sum_{t=1}^{n} \ell_t(h) \right) \\
&= \sup_{h \in \mathcal{H}} \min_{h' \in H_{n'}} \sum_{t=1}^{n} \left(\ell_t(h') - \ell_t(h) \right) \\
&\overset{\text{(a)}}{\leq} \sup_{h \in \mathcal{H}} \min_{h' \in H_{n'}} n\Delta_n(h, h') \\
&\overset{\text{(b)}}{\leq} \sup_{h \in \mathcal{H}} \min_{h' \in H_{n'}} n\Delta(h, h') + n\varepsilon_n && \text{w.p. } 1-\beta' \\
&\overset{\text{(c)}}{\leq} \sup_{h \in \mathcal{H}} \min_{h' \in H_{n'}} n\Delta_{n'}(h, h') + n\varepsilon_{n'} + n\varepsilon_n && \text{w.p. } 1-2\beta' \\
&\overset{\text{(d)}}{\leq} 0 + n\varepsilon_{n'} + n\varepsilon_n \\
&\overset{\text{(e)}}{\leq} 4n\sqrt{2\frac{2d\log\frac{en'}{d} + \log\frac{4}{\beta'}}{n}} \\
&\overset{\text{(f)}}{\leq} 4\sqrt{4nd\log\frac{en'}{d}} + 4\sqrt{2n\log\frac{4}{\beta'}},
\end{aligned}
$$

(a) Two hypotheses have a different loss whenever their prediction is different, thus we use the definition of the empirical disagreement in (7).
(b)-(c) We apply two times Lemma 3.
(d) The minimum disagreement $\Delta_{n'}(h, h')$ is zero for any hypothesis $h \in \mathcal{H}$. In fact, since $H_{n'}$ is built according to the same inputs $(x'_1, \dots, x'_n)$ on which $\Delta_{n'}(h, h')$ is measured, it is always possible to find a hypothesis $h' \in H_{n'}$ with exactly the same sequence of predictions as any $h \in \mathcal{H}$.
(e) By assumption $n' \geq n$ and from the definition of ε_n and $\varepsilon_{n'}$ in Lemma 3.
(f) We apply $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ to make the bound similar to the bound for the EWF.

By joining the bound for R_{EWF} and R_H, and by setting $\beta = 3\beta'$ we obtain

$$
R_n \leq \sqrt{\frac{nd}{2}\log\frac{en'}{d}} + \sqrt{\frac{n}{2}\log\frac{1}{\beta'}} + 4\sqrt{8\frac{nd}{2}\log\frac{en'}{d}} + 4\sqrt{4\frac{n}{2}\log\frac{4}{\beta'}}
$$

Algorithm 1 The Epoch-based Stochastic Adversarial (EStochAd) forecaster

Input: hypothesis set $\mathcal{H}$
Initialize: $H_0 = \emptyset$ with any $h \in \mathcal{H}$
for $k = 0, 1, 2, \ldots$ **do**
  Set $t_k = 2^k$, $t_{k+1} = 2^{k+1}$, $N_k = |H^{(k)}|$, and $\eta_k = \sqrt{2 \log N_k / n_k}$
  Initialize $w_i^{t_k} = 1, \;\; i \in \{1, \ldots, N\}$
  for $t = t_k$ to $t_{k+1} - 1$ **do**
    Observe x_t
    Sample $h_t \sim \mathbf{p}^t$, with $p_i = w_i^t / (\sum_{j=1}^{N_k} w_j^t)$
    Predict $\widehat{y}_t = h_t(x_t)$
    Observe the true label y_t
    Update weights $w_j^{t+1} = w_j^t \exp\left(-\eta_k \ell_t(h_j)\right), \;\; j \in \{1, \ldots, N_k\}$
  end for
  Build H_{k+1} according to inputs $\{x_1, \ldots, x_{t_{k+1}-1}\}$
end for

and the final statement follows by setting $c_1 = 1 + 8\sqrt{2}$ and $c_2 = 9$. □

3.3. The Epoch-based Stochastic Adversarial (EStochAd) Forecaster

In the previous section we assumed a sequence of inputs $(x'_1, \ldots, x'_n)$ could be sampled from P before starting the learning process. However, this assumption is often unrealistic when the distribution P is unknown and inputs are revealed only during the learning process. In this section we devise an epoch-based algorithm in which the hypothesis set is incrementally built in epochs according to the inputs experienced so far.

The algorithm works is epochs such that epoch k is $n_k = t_{k+1} - t_k$ steps long, from time $t = t_k$ to $t_{k+1} - 1$. At the beginning of epoch k, a grid $H^{(k)}$ is build on the basis of the sequence of inputs $\{x_t\}_{t=1}^{t_k - 1}$ and a EWF is run on $H^{(k)}$ until the end of epoch k. The resulting algorithm is summarized in Algorithm 1. As it can be noticed, EStochAd is an anytime algorithm since the time horizon n does not need to be known in advance. In fact, at epoch k the learning parameter η_k is set as in the EWF according to the length n_k of the epoch, independently from the value of n.

According to Theorem 2, whenever $t_k \geq n_k$ the regret of an EWF with

experts in $H^{(k)}$ and parameter $\eta_k = \sqrt{2 \log N_k / n_k}$ in epoch k is

$$R^{(k)} = \sum_{t=t_k}^{t_{k+1}-1} \ell_t(h_t) - \inf_{h \in \mathcal{H}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(h)$$
$$\leq c_1 \sqrt{n_k d \log \frac{e t_k}{d}} + c_2 \sqrt{\frac{n_k}{2} \log \frac{12}{\beta}} \tag{9}$$

with probability $1 - \beta$.

The next theorem shows that if the length of each epoch is set properly, then the regret of the EStochAd algorithm is bounded by $O\left(\sqrt{nd \log n}\right)$ with high probability.

Theorem 3. *Let the length of the epochs be $n_k = 2^k$, thus $t_k = 2^k$. At the beginning of epoch k a hypothesis set $H^{(k)}$ is built according to all the inputs up to time $t_k - 1$ and the weight of each hypothesis is initialized to 1. Let $\mathcal{H}$ be a hypothesis space with finite VC-dimension $d = VC(\mathcal{H}) < \infty$. For any $n > 0$, the EStochAd algorithm described above satisfies*

$$R_n \leq c_3 \sqrt{nd \log \frac{en}{d}} + c_4 \sqrt{n \log \frac{12(\lfloor \log_2 n \rfloor + 1)}{\alpha}} \tag{10}$$

with probability $1 - \alpha$, where $c_3 = 18 + 17\sqrt{2}$ and $c_4 = 9(2 + \sqrt{2})$.

Proof. The theorem directly follows from Theorem 2 and from the definition of epochs. Given $t_k = n_k = 2^k$ the regret for each epoch can be rewritten as

$$R^{(k)} \leq c_1 \sqrt{2^k d \log \frac{e 2^k}{d}} + c_2 \sqrt{\frac{2^k}{2} \log \frac{12}{\beta}}$$

Let $K = \lfloor \log_2 n \rfloor + 1$ be the index of the epoch containing the step n and $t_K = \min(2^K, n+1)$. The total regret over all the K epochs can be bounded

as follows

$$
\begin{aligned}
R_n &= \sum_{t=1}^{n} \ell_t(h) - \inf_{h\in\mathcal{H}} \sum_{t=1}^{n} \ell_t(h) \\
&= \sum_{k=0}^{K-1} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(h) - \inf_{h\in\mathcal{H}} \sum_{k=0}^{K-1} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(h) \\
&\overset{\text{(a)}}{\leq} \sum_{k=0}^{K-1} \left(\sum_{t=t_k}^{t_{k+1}-1} \ell_t(h) - \inf_{h\in\mathcal{H}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(h) \right) \\
&\leq \sum_{k=0}^{K-1} R^{(k)} = \sum_{k=0}^{\lfloor \log_2 n \rfloor} R^{(k)} \\
&\overset{\text{(b)}}{\leq} \left(c_1 \sqrt{d \log \frac{en}{d}} + c_2 \sqrt{\frac{1}{2} \log \frac{12}{\beta}} \right) \sum_{k=0}^{\lfloor \log_2 n \rfloor} \sqrt{2^k} \quad \text{w.p. } 1 - \beta(\lfloor \log_2 n \rfloor + 1) \\
&\leq \left(c_1 \sqrt{d \log \frac{en}{d}} + c_2 \sqrt{\frac{1}{2} \log \frac{12}{\beta}} \right) \frac{\sqrt{2n} - 1}{\sqrt{2} - 1} \\
&\overset{\text{(c)}}{\leq} c_3 \sqrt{nd \log \frac{en}{d}} + c_4 \sqrt{n \log \frac{12}{\beta}}.
\end{aligned}
$$

(a) the regret is upper-bounded by considering the best hypothesis in each epoch rather than on the whole horizon of n steps.
(b) The inner term in the summation is the regret for epoch k and is bounded as in (9).
(c) Constants are obtained by setting $c_3 = c_1(2 + \sqrt{2})$ and $c_4 = c_2(2 + \sqrt{2})$.

Finally, by using a union bound and setting $\alpha = \beta(\lfloor \log_2 n \rfloor + 1)$ the result is obtained from the definition of the length of each epoch and some algebra. □

We postpone a detailed analysis of this result and a comparison with other existing results to Section 6.

It is worth noting that, from an implementation point of view the set of hypotheses $H^{(k)}$ does not need to be regenerated from scratch at the beginning of each epoch k but it can be built incrementally as new inputs comes in. As a consequence, for each hypothesis h_i already available at the previous epoch, its weight w_i is initialized according to the cumulative loss up

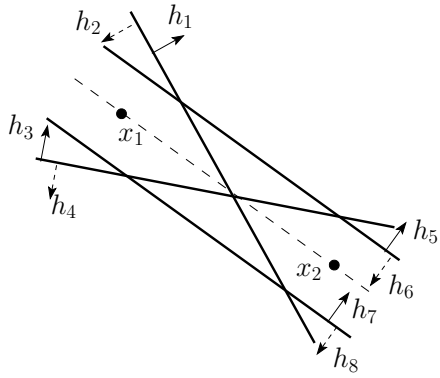

Figure 4: In a two-dimensional binary classification problem, each pair of input points can be classified in four different ways using linear classifiers. The arrow indicates the half-space which is positively labeled.

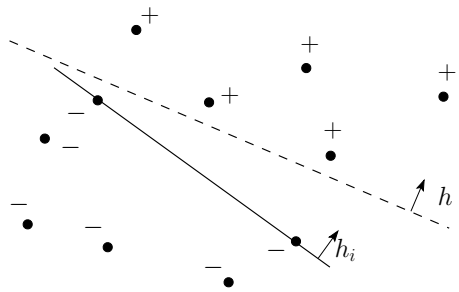

Figure 5: Example of the procedure to follow to find the hypothesis h_i in $H^{(k+1)}$ having the same sequence of predictions as any $h \in \mathcal{H}$.

to end of the previous epoch. Similarly, new hypotheses can inherit the weight of hypotheses belonging to the same class before the refinement. Although no improvement in the bound can be proved, using the past performance to initialize the weight for new hypotheses is likely to have a positive impact in the actual performance.

4. Computational Complexity

Although the main focus of this paper is the introduction of the hybrid stochastic-adversarial setting and the analysis of its learnability, in this section we discuss the efficiency of the proposed algorithm. At each epoch k EStochAd is divided into two phases: a learning phase in which a EWF is run on N_k experts and a phase in which the set of experts is updated accord-

Algorithm 2 Algorithm to build a grid $H^{(k+1)}$ at the end of epoch k after observing $t_k - 1$ inputs (sketch).

Input: linear classifiers, $\{x_t\}_{t=1}^{t_k-1}$ inputs, $\mathcal{X} = \mathbb{R}^d$
Initialize: $H^{(k+1)} = \emptyset$
for all d-tuple $\{x_i\}_{i=1}^{d}$ of inputs from $\{x_t\}_{t=1}^{t_k-1}$ **do**
  Build the hyper-plane h passing through $\{x_i\}_{i=1}^{d}$
  for $i = 1$ to 2^d **do**
    Transform h and generate the hyper-plane h'
    Define the classifiers h_{2i-1} and h_{2i} equal to h' but with two different directions
  end for
  Add $\{h_j\}_{j=1}^{2\cdot 2^d}$ to $H^{(k+1)}$
end for

ing to the inputs observed so far. By (Sauer, 1972)'s Lemma the number of experts N_k is at most $O(t_k^d)$ where d is the VC-dimension of $\mathcal{H}$. As a result, the computational complexity of the learning phase is polynomial in the time horizon n and exponential in d. In fact, at each round t the EWF updates the weights of each of the N_k experts according to the loss they incur at time t. Thus the computational cost of each epoch k is $O(n_k t_k^d)$ where n_k is the length of the epoch. By setting $n_k = t_k = 2^k$ and $K = \lfloor \log_2 n \rfloor + 1$, we obtain that learning phase has a total computation cost of order $O(n^{d+1})$. The critical part now is to show whether it is possible to build the grid $H^{(k)}$ on the basis of the inputs $\{x_t\}_{t=1}^{t_{k-1}}$ in an efficient way. The method to generate $H^{(k)}$ highly depends on the specific hypothesis space $\mathcal{H}$ at hand. In the following we propose a method for linear spaces with a polynomial complexity in the time horizon n.

Let the input space $\mathcal{X} = \mathbb{R}^d$ and t_k the number of input points observed so far. Any tuple of d input points can be classified in at most 2^d ways. A set of linear classifiers generating all the possible combinations can be easily obtained by computing the hyper-plane passing through the d inputs first [2] and then transforming it through appropriate infinitesimal (i.e., without intersecting other inputs) translations and rotations in order to obtained all the

[2] All the methods to compute the hyper-plane in $\mathbb{R}^d$ passing through d points have a polynomial complexity in d.

2^d possible classifications (see Figure 4 for an example in two dimensions). For each of the 2^d combinations we generate two classifiers, one for each of the two possible directions. This process should be repeated for any possible combination of d inputs among $\{x_t\}_{t=1}^{t_k-1}$. Let A be the cost of computing the hyper-plane and B the cost of each transformation on the hyper-plane, the total computational cost of the generation of $H^{(k+1)}$ at the end of epoch k is

$$\text{cost}(H_{k+1}) = \binom{t_k}{d} \left(A + 2^d 2B\right).$$

Using the bound on the binomial coefficient $\binom{t_k}{d} \leq \left(\frac{et_k}{d}\right)^d$, $t_k = 2^k$, and taking the sum over the number of epochs, we obtain that the computational complexity of the update phase of the grid $H^{(k)}$ over epochs is of order $O((A+2^d 2B)n^d)$. Although this method may generate redundant hypotheses (i.e., hypotheses having the same classification on points $\{x_t\}_{t=1}^{t_k-1}$), its complexity is not worse than for the learning phase with the EWF, thus making the overall complexity of EStochAd with linear classifiers polynomial in n with exponent the VC-dimension d.

In the following lemma we prove that the procedure outlined in Algorithm 2 generates a grid $H^{(k+1)}$ containing hypotheses with the same predictions on inputs $\{x_t\}_{t=1}^{t_k-1}$ as any hypothesis in $\mathcal{H}$.

Lemma 4. *For any $h \in \mathcal{H}$ there exist a hypothesis $h_i \in H^{(k+1)}$, where the grid $H^{(k+1)}$ is built according to Algorithm 2, such that the sequence of prediction of h and h_i is the same on $\{x_t\}_{t=1}^{t_k-1}$.*

Proof. The proof follows by construction of $H^{(k+1)}$. Let h be any linear classifier in $\mathcal{H}$ (see Figure 5 for an example in 2-d). It is always possible to apply to h a transformation so that the resulting classifier h' still has the same classification on the t_k inputs and it passes through a subset of d points among $\{x_t\}_{t=1}^{t_k-1}$. Since Algorithm 2 enumerates all the possible hyper-planes passing through d points, h' is used as a base classifier to generate the classifiers in $H^{(k+1)}$. Thus, there always exists among the classifiers generated from h' one having exactly the same classification as h. □

5. Extensions

In this section, we discuss possible extensions of the proposed algorithm to different settings.

5.1. Multi-Label Classification

Although we analyzed the performance of EStochAd in the case of binary classification, the extension to the case of multi-label classification is straightforward. In order to measure the complexity of $\mathcal{H}$ we refer to the extension to multi-label classification of the VC-dimension proposed by Natarajan (1989) [3]. The resulting algorithm is the same as in Algorithm 1 and still runs an EWF on a grid $H^{(k)}$ obtained by partitioning the space $\mathcal{H}$ into classes of hypotheses with the same sequence of predictions on past inputs.

Theorem 4. *For any $n > 0$, let $m > 0$ be number of labels and $\mathcal{H}$ a hypothesis with finite Natarajan dimension $d = Ndim(\mathcal{H}) < \infty$. The EStochAd algorithm satisfies*

$$R_n \leq c_5 \sqrt{nd \log \frac{enm^2}{2d}} + c_6 \sqrt{n \log \frac{3 \log_2 n}{\alpha}}, \tag{11}$$

with probability $1 - \alpha$, with a universal constants c_5 and c_6.

Proof. The proof follows the same steps as in Theorem 3. The main difference is that the number of hypotheses in H_n is now bounded by $|H_n| \leq (\frac{enm^2}{2d})^d$ (Ben-David et al., 1995) and that in Lemma 3 the $Ndim(\mathcal{H})$ is used instead of the VC-dimension. □

5.2. Bandit Information

In the protocol in Figure 2 at the end of each episode the true label chosen by the adversary is explicitly revealed to the learner, thus defining a *full* information classification problem. However, in many applications (e.g., web advertisement systems) only the loss corresponding to the chosen hypothesis (*bandit* feedback) is available to the learner.

The EStochAd algorithm can be extended to solve the hybrid stochastic-adversarial classification problem with bandit information simply by substituting the EWF with a bandit algorithm such as Exp4 (Auer et al., 2003), thus defining the so-called Bandit-EStochAd algorithm (Algorithm 3). Let us consider the more general case illustrated in Figure 1 in which instead of selecting a label, at each round t the adversary chooses a bounded loss function $\ell_t : \mathcal{Y} \to [0, 1]$. In the prediction phase, the original EStochAd

[3] For more details about complexity measures for m-values functions, we refer to Ben-David et al. (1995).

Algorithm 3 The Bandit-EStochAd forecaster

Input: hypothesis set $\mathcal{H}$
Initialize: $H_0 = \emptyset$ with any $h \in \mathcal{H}$
for $k = 0, 1, 2, \ldots$ **do**
  Set $t_k = 2^k$, $t_{k+1} = 2^{k+1}$, $N_k = |H^{(k)}|$, and $\eta_k = \sqrt{2 \log N_k / n_k}$
  Initialize $w_i^{t_k} = 1, \;\; i \in \{1, \ldots, N\}$
  for $t = t_k$ to $t_{k+1} - 1$ **do**
    Observe x_t
    Sample $h_t \sim \mathbf{p}^t$, with $p_i = w_i^t / (\sum_{j=1}^{N_k} w_j^t)$
    Sample $\widehat{y}_t \sim \mathbf{q}^t$, where

$$q_j^t = (1-\gamma)\mathbb{I}\left\{j = h_t(x_t)\right\} + \frac{\gamma}{m}, \;\; j \in \{1, \ldots, m\}$$

    Receive loss $\ell(\widehat{y}_t)$
    Define $\hat{\ell}_t(h_i) = \frac{\ell(\widehat{y}_t)}{q_{\widehat{y}_t}^t} \mathbb{I}\left\{h_i(x_t) = \widehat{y}_t\right\}, \;\; i \in \{1, \ldots, N_k\}$
    Update weights $w_i^{t+1} = w_i^t \exp\left(-\eta_k \hat{\ell}_t(h_i)\right)$
  end for
  Build H_{k+1} according to inputs $\{x_1, \ldots, x_{t_{k+1}-1}\}$
end for

algorithm is used to select a hypothesis h_i in the grid $H^{(k)}$ built so far. According to Exp4 an additional randomization over all the possible m labels is introduced and a prediction $\widehat{y}_t$ is returned. At the end of each round t, the learner incurs a loss $\ell_t(h_t(x_t))$ (the only information available) for which an unbiased estimate of the loss $\hat{\ell}_t(h)$ for any hypothesis h is built. Finally, the weights of the hypotheses in H_t are updated according to EStochAd. We can prove the following regret bound for Bandit-EStochAd.

Theorem 5. *Let $m > 0$ be the number of arms (i.e., labels), $\mathcal{H}$ be a hypothesis set with finite Natarajan dimension $d = Ndim(\mathcal{H}) < \infty$, and ℓ be bounded in $[0, 1]$. For any $n > 0$, the Bandit-EStochAd algorithm satisfies*

$$R_n \leq O\left(\sqrt{nmd \log \frac{nm^2}{\alpha}} + d \log \frac{nm^2}{\alpha}\right), \tag{12}$$

with probability $1 - \alpha$.

Proof. In Theorem 2 the first part of the regret of EStochAd can be immediately derived from the bandit algorithm working on the set H_n. For instance, for Exp4 with N experts and m labels it is possible to prove the high-probability regret bound

$$R_n(\text{Exp4}) \leq 4\sqrt{nm \log \frac{nN}{\beta}} + 8 \log \frac{nN}{\beta},$$

with probability $1 - \beta$. As discussed in the previous section, in case of m labels the number of experts at time n is bounded by $N = |H_n| \leq (\frac{enm^2}{2d})^d$. Besides, the second term in (6) is not affected by the different feedback in full and bandit settings and remains unchanged. The only difference is in step *(a)* of Theorem 2. Indeed, when two hypotheses have the same prediction their loss is the same. On the other hand, if the predictions are different, the difference between the losses cannot be greater than 1. Thus, $\ell_t(h) - \ell_t(h') \leq \mathbb{I}\{h(x_t) \neq h'(x_t)\}$. As a result, the leading term in the cumulative regret is due to Exp4 and the statement follows. □

5.3. Regression

So far we considered classification problems in which $\mathcal{H}$ is a discrete-valued space of functions, we now show that for some loss functions that analysis can be easily extended to the regression setting. We first recall the definition of pseudo-dimension of a space of real-valued functions.

Definition 1. *Let $\mathcal{F}$ be a space of bounded real-valued functions $f : \mathcal{X} \to [0, B]$ and $\{x_t\}_{t=1}^n$ be a set of points in $\mathcal{X}$. We say that the points $\{x_t\}_{t=1}^n$ are pseudo-shattered by $\mathcal{F}$ if there are $\{y_t\}_{t=1}^n \in [0, B]^n$ such that for any $\mathbf{b} \in \{0, 1\}^n$, there is a $f_{\mathbf{b}} \in \mathcal{F}$ such that*

$$f_{\mathbf{b}}(x_t) \geq y_t \Leftrightarrow b_t = 1, \quad 1 \leq t \leq n. \tag{13}$$

The largest n such that there exists a set $\{x_t\}_{t=1}^n$ pseudo-shattered by $\mathcal{F}$ is the pseudo-dimension of $\mathcal{F}$, denoted by $V_{\mathcal{F}}^+$.

We also need the definition of $(\epsilon, 1)$-covering number of $\mathcal{F}$.

Definition 2. *Let $\mathcal{F}$ be a space of bounded real-valued functions $f : \mathcal{X} \to [0, B]$. Every finite collection of functions $f_1, \ldots, f_N \in \mathcal{F}$ is a $(\epsilon, 1)$-cover of $\mathcal{F}$ if for any $f \in \mathcal{F}$ there exist a f_i $(1 \leq i \leq N)$ such that*

$$||f - f_i||_1 \leq \epsilon. \tag{14}$$

The $(\epsilon, 1)$-covering number of $\mathcal{F}$ is the smallest N such that $f_1, \ldots, f_N$ is a $(\epsilon, 1)$-cover of $\mathcal{F}$, and we denote it by $\mathcal{N}_1(\epsilon, \mathcal{F})$.

Similar, let $\{x_t\}_{t=1}^n$ be a set of points in $\mathcal{X}$, a collection of functions $f_1, \ldots, f_N \in \mathcal{F}$ is an empirical $(\epsilon, 1)$-cover of $\mathcal{F}$ on $\{x_t\}_{t=1}^n$ if for any $f \in \mathcal{F}$ there exist a f_i $(1 \leq i \leq N)$ such that

$$\frac{1}{n}\sum_{t=1}^{n} |f(x_t) - f_i(x_t)| \leq \epsilon. \tag{15}$$

The $(\epsilon, 1)$-covering number of $\mathcal{F}$ is the smallest N such that $f_1, \ldots, f_N$ is an empirical $(\epsilon, 1)$-cover of $\mathcal{F}$ on $\{x_t\}_{t=1}^n$, and we denote it by $\mathcal{N}_1(\epsilon, \mathcal{F}, \{x_t\}_{t=1}^n)$.

Finally, we recall Pollard's inequality (Pollard, 1984).

Lemma 5. *Let $\mathcal{F}$ a set of functions $f : \mathcal{X} \to [0, B]$, and $x_1, \ldots, x_n$ be a sequence of i.i.d. samples from a distribution P. For any $n > 0$, $\epsilon > 0$ then*

$$\mathbb{P}\left[\sup_{f\in\mathcal{F}}\left|\frac{1}{n}\sum_{t=1}^{n} f(x_t) - \mathbb{E}[f(x_1)]\right| \geq \epsilon\right] \leq 8\mathbb{E}\left\{\mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{F}, x_1^n\right)\right\}\exp\left(-\frac{n\epsilon^2}{128B^2}\right).$$

Equivalently, let $V = V_{\mathcal{F}}^{+}$ be the pseudo-dimension of $\mathcal{F}$, then with probability $1 - \beta$ for all $f \in \mathcal{F}$

$$\left|\frac{1}{n}\sum_{t=1}^{n} f(x_t) - \mathbb{E}[f(x_1)]\right| \leq \epsilon = 8B\sqrt{2\frac{\Lambda(n, V, \beta)}{n}}, \tag{16}$$

where $\Lambda(n, V, \beta) = V \log n + \log \frac{e}{\beta} + \log\left(24(\frac{9}{2}e^2)^V\right)$.

Proof. We just report the proof of the second statement. It is sufficient to show that

$$8\mathbb{E}\left\{\mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{F}, x_1^n\right)\right\}\exp\left(-\frac{n\epsilon^2}{128B^2}\right) \leq \beta \tag{17}$$

for the ϵ in (16). We first notice that $\Lambda(n, V, \beta) > 1$ and $\epsilon > 8B\sqrt{\frac{2}{n}}$. Using the bound for covering numbers in Haussler (1992), the definition of

$\Lambda(n, V, \beta)$, $a = 24eB$, $b = (128B^2)^{-1}$, and some algebra we obtain

$$\begin{aligned}
&8\mathbb{E}\left\{\mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{F}, x_1^n\right)\right\} \exp\left(-\frac{n\epsilon^2}{128B^2}\right) \\
&\leq 24\left(\frac{a^2}{\epsilon^2}\right)^V \exp(-bn\epsilon^2) \\
&= 24(a^2bn)^V \exp(\Lambda(n, V, \beta)) \\
&= 24(a^2b)^V n^V n^{-V} \frac{\beta}{e} \frac{1}{24(a^2b)^V} \leq \beta.
\end{aligned}$$

□

We are now ready to define the extension of EStochAd to regression. We consider the general full information setting in which at round t the adversary chooses a bounded loss function $\ell_t : \mathcal{Y} \to [0, M]$, with $\mathcal{Y} = [0, B]$, and the learner chooses a function $f_t \in \mathcal{F}$. At the end of the round the learner incurs a loss $\ell_t(f_t(x_t)$ depending on the input x_t drawn from a distribution P. Finally, the loss function $\ell_t(\cdot)$ is revealed to the learner. Similar to the classification case, the objective of the learner is to minimize the regret

$$R_n = \sum_{t=1}^{n} \ell_t(f_t(x_t)) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{n} \ell_t(f(x_t)).$$

The structure of the algorithm is mostly the same as in Algorithm 1. Instead of a discrete-valued set of hypotheses $\mathcal{H}$, we consider the space $\mathcal{F}$ of real-valued functions bounded in $[0, B]$. At the end of each epoch the grid of hypotheses $H^{(k)}$ is substituted by an $(\epsilon, 1)$-cover of $\mathcal{F}$ on inputs $\{x_1, \ldots, x_{t_k-1}\}$, denoted by $F^{(k)}$. It is possible to prove that this algorithm achieves the same performance as EStochAd in classification under suitable assumptions on the loss function making it possible to define a relationship between the space of loss functions and $\mathcal{F}$. In particular, we prove the following.

Theorem 6. *Let $\mathcal{F}$ be a space of bounded real-valued functions $f : \mathcal{X} \to [0, B]$ with finite pseudo dimension $V = V_{\mathcal{F}}^{+} < \infty$. At each round t the adversary chooses a loss function ℓ_t which is Lipschitz with constant L. The EStochAd algorithm for regression described above satisfies*

$$R_n \leq c_7 LB \left(\sqrt{nV \log n} + \sqrt{n \log \frac{12e(\lfloor \log_2 n \rfloor + 1)}{\alpha}} + \sqrt{n \log \left[24\left(\frac{9}{2}e^2\right)^V\right]}\right) \tag{18}$$

with probability $1-\alpha$*, where* $c_7 = 49(\sqrt{2}-1)$*.*

Proof. First we need to derive the equivalent of Theorem 2 in case of regression to bound the regret at each epoch. The regret $R^{(k)}$ can be decomposed as

$$\begin{aligned}
R^{(k)} &= \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f_t) - \inf_{f\in\mathcal{F}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f) \\
&= \left(\sum_{t=t_k}^{t_{k+1}-1} \ell_t(f) - \min_{f'\in F^{(k)}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f') \right) + \left(\min_{f'\in F^{(k)}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f') - \inf_{f\in\mathcal{F}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f) \right) \\
&= R_{EWF}^{(k)} + R_F^{(k)}
\end{aligned}$$

where $F^{(k)}$ is an $(\epsilon, 1)$-cover of $\mathcal{F}$ on inputs $\{x_1, \dots, x_{t_k-1}\}$. The first component of the regret can still be bounded using the bound of the EWF. Since $F^{(k)}$ is an $(\epsilon, 1)$-cover of $\mathcal{F}$, it contains $N_k = \mathcal{N}_1\left(\epsilon, \mathcal{F}, x_1^{t_k-1}\right)$ experts. By Haussler (1992), the following bound on the covering number holds

$$\mathcal{N}_1\left(\epsilon, \mathcal{F}, x_1^{t_k-1}\right) \le 3\left(\frac{2eB}{\epsilon}\log\frac{3eB}{\epsilon}\right)^V \le 3\left(\frac{3eB}{\epsilon}\right)^{2V}.$$

Let $\epsilon = 8B\sqrt{\frac{2\Lambda(t_k,V,\beta)}{t_k}}$, as in Lemma 5 we notice that $\epsilon \ge 8B\sqrt{\frac{2}{t_k}}$, thus the number of experts can be bounded as

$$N_k \le 3\left(\frac{9e^2}{128}t_k\right)^V.$$

Thus, we can bound the first term $R_{EWF}^{(k)}$ in the regret $R^{(k)}$ by the performance of the EWF on N_k experts

$$R^{(k)} \le LB\left(\sqrt{\frac{2^k}{2}\log N_k} + \sqrt{\frac{2^k}{2}\log\frac{1}{\beta}}\right) + R_F^{(k)},$$

where the multiplicative term LB is the bound over the loss function. In

order to bound the second term we follow similar steps as in Theorem 2.

$$
\begin{aligned}
R_F^{(k)} &= \left(\min_{f' \in F^{(k)}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f') - \inf_{f \in \mathcal{F}} \sum_{t=t_k}^{t_{k+1}-1} \ell_t(f) \right) \\
&= \sup_{f \in \mathcal{F}} \min_{f' \in F^{(k)}} \sum_{t=t_k}^{t_{k+1}-1} \left(\ell_t(f') - \ell_t(f) \right) \\
&\overset{(a)}{\leq} \sup_{f \in \mathcal{F}} \min_{f' \in F^{(k)}} \sum_{t=t_k}^{t_{k+1}-1} L \left(f'(x_t) - f(x_t) \right) \\
&\overset{(b)}{\leq} \sup_{f \in \mathcal{F}} \min_{f' \in F^{(k)}} 2^k L \mathbb{E}_{x \sim P} \left[f'(x) - f(x) \right] + 2^k L \epsilon \qquad \text{w.p. } 1 - \beta' \\
&\overset{(c)}{\leq} L \sup_{f \in \mathcal{F}} \min_{f' \in F^{(k)}} 2^k L \sum_{t=1}^{t_k - 1} \left(f'(x_t) - f(x_t) \right) + 2L \cdot 2^k \epsilon \qquad \text{w.p. } 1 - 2\beta' \\
&\overset{(d)}{\leq} 3L \cdot 2^k \epsilon = 24 LB \sqrt{2 \frac{\Lambda(2^k, V, \beta)}{2^k}}.
\end{aligned}
$$

(a) By assumption that the loss is Lipschitz.
(b)-(c) We apply two times Lemma 5.
(d) The grid $F^{(k)}$ is obtained by building an ϵ-cover of $\mathcal{F}$ on the samples $\{x_t\}_{t=1}^{t_k-1}$. Thus, by definition 2 the closest function $f' \in F^{(k)}$ to any function $f \in \mathcal{F}$ is not further than ϵ.

By joining the bound for $R_{EWF}^{(k)}$ and $R_F^{(k)}$ and some simplification, we obtain

$$
R^{(k)} \leq \frac{49}{2} LB\sqrt{2} \left(\sqrt{2^k V \log 2^k} + \sqrt{2^k \log \frac{e}{\beta'}} + \sqrt{2^k \log \left[24 \left(\frac{9}{2} e^2 \right)^V \right]} \right).
$$

Finally, we follow the same steps as in Theorem 3 and we obtain the final statement. □

It is interesting to notice that the class of Lipschitz losses includes commonly used loss functions such as L_1 and squared loss. Let $y_t \in [0, B]$ be the output at time t, the L_1 loss is defined as $\ell_t(y) = |y - y_t|$. It is immediate to

for $t = 1, 2, \ldots$ **do**
Simultaneously,

- A stochastic input x_t is sampled i.i.d. from P
- Player A selects strategy $h_{A,t}$
- Player B selects strategy $h_{B,t}$

Player A (resp., B) plays action $\widehat{y}_{A,t} = h_{A,t}(x_t)$ (resp., $\widehat{y}_{B,t} = h_{B,t}(x_t)$)
Return feedback

- $\ell_A(\widehat{y}_{A,t}, \widehat{y}_{B,t}, x_t)$ and $\ell_B(\widehat{y}_{A,t}, \widehat{y}_{B,t}, x_t)$ (*bandit* information)
- *or* $\ell_A(\cdot, \widehat{y}_{B,t}, x_t)$ and $\ell_B(\widehat{y}_{A,t}, \cdot, x_t)$ (*full* information)

Player A (resp., B) incurs a loss $\ell_A(\widehat{y}_{A,t}, \widehat{y}_{B,t}, x_t)$ (resp., $\ell_B(\widehat{y}_{A,t}, \widehat{y}_{B,t}, x_t)$)
end for

Figure 6: The two-player strategic repeated game with stochastic side information.

verify that this loss is a Lipschitz function with $L = 1$ and $M = B$. In case of a squared loss $\ell_t(y) = (y - y_t)^2$ we have

$$\begin{aligned}|\ell_t(y_1) - \ell_t(y_2)| &= |(y_1 - y_t)^2 - (y_2 - y_t)^2| \\ &= |y_1^2 - y_2^2 - 2y_t(y_1 - y_2)| = |(y_1 - y_2)(y_1 + y_2 - 2y_t)| \\ &\leq 2B|y_1 - y_2|.\end{aligned}$$

Thus the squared loss is a Lipschitz functions with $M = B^2$ and $L = 2B$.

5.4. Application in Games

In this section we consider an extension of the stochastic-adversarial prediction problem to a two-player strategic repeated game with stochastic side information. Like in the general problem illustrated in Figure 1, the game could be either *full* or *bandit* information, depending on whether at the end of each round the learners receive the loss function $\ell_A(\cdot, \widehat{y}_{B,t}, x_t)$ (resp. $\ell_B(\widehat{y}_{A,t}, \cdot, x_t)$) or only the loss they incurred. Our main contribution here is to show that in the case of a zero-sum game, if both players play according to the (Bandit-)EStochAd algorithm, then the empirical frequencies of the strategies converge to the set of Nash equilibria.

For sake of simplicity we consider the same set of strategies for both players. Let A and B be two players and $\mathcal{H}$ be the set of strategies h mapping an input $x \in \mathcal{X}$ to an action in $\mathcal{Y} = \{1, \ldots, m\}$. The repeated game between player A and B is sketched in Figure 6. At each round t, an input x_t is drawn from P and, simultaneously, the players select strategies $h_{A,t} \in \mathcal{H}$ and $h_{B,t} \in \mathcal{H}$. As a result, they incur losses $\ell_A(h_{A,t}(x_t), h_{B,t}(x_t), x_t)$ and $\ell_B(h_{A,t}(x_t), h_{B,t}(x_t), x_t)$ respectively ($\ell_{A,t}(h_{A,t})$ and $\ell_{B,t}(h_{B,t})$ for short in the following). We define the expected loss for player A with respect to the input distribution P as

$$\overline{\ell}_A(h_A, h_B) = \mathbb{E}_{x \sim P}\left[\ell_A(h_A(x), h_B(x), x)\right].$$

Let $\mathcal{D}(\mathcal{H})$ be the set of distributions on the set of pure strategies $\mathcal{H}$. Given mixed strategies σ_A and σ_B in $\mathcal{D}(\mathcal{H})$ we define its corresponding expected loss (similarly for player B):

$$\overline{\ell}_A(\sigma_A, \sigma_B) = \mathbb{E}_{h_A \sim \sigma_A, h_B \sim \sigma_B}\left[\overline{\ell}_A(h_A, h_B)\right].$$

We say that a pair of strategies (σ_A^*, σ_B^*) is a Nash equilibrium if

$$\begin{aligned} \overline{\ell}_A(\sigma_A^*, \sigma_B^*) &\leq \overline{\ell}_A(\sigma_A, \sigma_B^*), \ \forall \sigma_A \in \mathcal{D}(\mathcal{H}) \\ \overline{\ell}_B(\sigma_A^*, \sigma_B^*) &\leq \overline{\ell}_B(\sigma_A^*, \sigma_B), \ \forall \sigma_B \in \mathcal{D}(\mathcal{H}). \end{aligned}$$

Now we consider the problem of approximating a Nash equilibrium in the zero-sum case (i.e., $\overline{\ell}_A(\cdot,\cdot) = -\overline{\ell}_B(\cdot,\cdot)$). In order to define the value of the game and apply the minimax theorem we need $\mathcal{D}(\mathcal{H})$ to be compact (Cesa-Bianchi & Lugosi, 2006). In the following, we assume that $\mathcal{H}$ is a compact metric space, which is a sufficient condition for $\mathcal{D}(\mathcal{H})$ to be compact (see e.g., Stoltz & Lugosi (2007)). Under this assumption, the minimax theorem (e.g., Cesa-Bianchi & Lugosi, 2006) holds

$$\begin{aligned} V &= \sup_{\sigma_B \in \mathcal{D}(\mathcal{H})} \inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \overline{\ell}_A(\sigma_A, \sigma_B) \\ &= \inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \sup_{\sigma_B \in \mathcal{D}(\mathcal{H})} \overline{\ell}_A(\sigma_A, \sigma_B), \end{aligned} \tag{19}$$

where V is the value of the game. The following theorem proves that if both players run either EStochAd or Bandit-EStochAd (in full information and bandit information respectively), then their performance converges to the value of the game and the empirical frequencies of their strategies converge to the set of Nash equilibria.

Theorem 7. *Let losses ℓ_A, ℓ_B be bounded in $[0,1]$, $\mathcal{H}$ be a compact metric set. If both players run (Bandit-)EStochAd in a zero-sum game with stochastic side information as defined above, then*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} \ell_A(h_{A,t}(x_t), h_{B,t}(x_t), x_t) = V \tag{20}$$

almost surely.

Proof. The proof is similar to the convergence proof for Hannan consistent strategies in zero-sum games (Cesa-Bianchi & Lugosi, 2006). We first prove the following

$$\lim_{n\to\infty} \frac{1}{n} \sum_{t=1}^{n} \ell_{A,t}(h_{A,t}) \leq V. \tag{21}$$

We note that the regret for both players can be bounded exactly as in (12). In fact, losses ℓ_A and ℓ_B are a special case of the adversarial loss function considered in Section 5.2. As a result, we have

$$\limsup_{n\to\infty} \left[\frac{1}{n} \sum_{t=1}^{n} \ell_{A,t}(h_{A,t}) - \inf_{h_A \in \mathcal{H}} \frac{1}{n} \sum_{t=1}^{n} \ell_{A,t}(h_A) \right] \leq 0, \tag{22}$$

with probability $1-\alpha$, where $\ell_{A,t}(h_A) = \ell_A(h_A, h_{B,t}, x_t)$. Let $g_A(h_{B,t}) = \ell_{A,t}(h_A) - \overline{\ell}_A(h_A, h_{B,t})$. By definition of the expected loss and by noticing that the hypothesis $h_{B,t}$ selected by the algorithm at time t does not depend on the input x_t, we have that $g_A(h_{B,t})$ for any $h_A \in \mathcal{H}$

$$\mathbb{E}_{x_t \sim P} \left[g_A(h_{B,t}) | \mathcal{F}_{t-1} \right] = 0,$$

where $\mathcal{F}_{t-1}$ is the σ-algebra generated by all random variables up to time $t-1$ (i.e., past inputs and hypotheses for both players A and B). Thus, $g_A(h_{B,1}), \cdots, g_A(h_{B,n})$ is a martingale difference sequence and we can apply Lemma 8. Thus we obtain that with probability $1-\beta$ for any function g_A induced by $h_A \in \mathcal{H}$, the empirical average $\frac{1}{n}\sum_{t=1}^{n} g_A(h_{B,t})$ asymptotically concentrates around 0. As a result, we have

$$\limsup_{n\to\infty} \left[\inf_{h_A \in \mathcal{H}} \frac{1}{n} \sum_{t=1}^{n} \ell_{A,t}(h_A) - \inf_{h_A \in \mathcal{H}} \frac{1}{n} \sum_{t=1}^{n} \overline{\ell}_A(h_A, h_{B,t}) \right] \leq 0. \tag{23}$$

Now, since the mapping $\sigma_A \mapsto \overline{\ell}_A(\sigma_A, h_{B,t})$ is linear, this function admits a pure strategy as minimum, and we have

$$\begin{aligned}\inf_{h_A \in \mathcal{H}} \frac{1}{n} \sum_{t=1}^{n} \overline{\ell}_A(h_A, h_{B,t}) &= \inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \frac{1}{n} \sum_{t=1}^{n} \overline{\ell}_A(\sigma_A, h_{B,t}) \\ &= \inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \overline{\ell}_A(\sigma_A, \sigma_B^n)\end{aligned}$$

where $\sigma_B^n(h) \in \mathcal{D}(\mathcal{H})$ is defined for any $h \in \mathcal{H}$ as $\sigma_B^n(h) = 1/n \sum_{t=1}^{n} \mathbb{I}\{h_{B,t} = h\}$. Finally, we have

$$\inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \overline{\ell}_A(\sigma_A, \sigma_B^n) \leq \sup_{\sigma_B \in \mathcal{D}(\mathcal{H})} \inf_{\sigma_A \in \mathcal{D}(\mathcal{H})} \overline{\ell}_A(\sigma_A, \sigma_B), \tag{24}$$

Putting together (22), (23), and (24) we obtain (21). The same result can be obtained for ℓ_B. From the assumption $\overline{\ell}_A(\cdot, \cdot) = -\overline{\ell}_B(\cdot, \cdot)$, minimax theorem (19), and since this result holds for any α, then we have (20) with probability 1. □

From the previous theorem and the compactness property of $\mathcal{D}(\mathcal{H})$ it also follows that the empirical frequencies of the mixed strategies σ_A^n and σ_B^n converge to the set of Nash strategies. Finally, it is interesting to notice that in the case of multi-label classification (in which the loss function $\ell_t(h) = \mathbb{I}\{h(x_t) \neq y_t\}$), the convergence rate (i.e., the regret per round) to the set of Nash equilibria is of the order $O(\sqrt{d/n \log(nm^2)})$ in the full information case, and $O(\sqrt{(md)/n \log(nm^2)})$ in the bandit information case.

6. Related Works

To the best of our knowledge this is the first work considering the online learning problem with stochastic inputs and adversarial labels. A similar setting is analyzed by Ryabko (2006) for batch supervised learning where the sequence of labels is adversarial and inputs are *conditionally* independent and identically distributed (i.e., inputs are drawn from distributions conditioned to labels). In particular, they show that in such a scenario many learning bounds (derived in the pure stochastic setting) remain unchanged. The main difference with the setting illustrated in this paper is that we considered the problem of online learning instead of batch learning and inputs are i.i.d. and not conditioned to labels.

Algorithm	**Setup**	**Hyp. space**	**Bound**	**Performance**
Empirical Risk Minimization	S/S	$VC(\mathcal{H}) < \infty$	HP-Regret	$\mathbb{E}_{(x,y)\sim P}\left[R_n\right] \leq \sqrt{nVC(\mathcal{H})\log n} + \sqrt{n\log\beta^{-1}}$
Exp. Weighted Forecaster (Cesa-Bianchi & Lugosi, 2006)	A/A	$\lvert\mathcal{H}\rvert = N < \infty$	HP-Regret	$R_n \leq \sqrt{n\log N} + \sqrt{n\log\beta^{-1}}$
Perceptron (Rosenblatt, 1958)	A/A	Linear	Mistake	$M_n \leq L + D + \sqrt{LD}$
Agnostic Online Learning (Ben-David et al., 2009)	A/A	$Ldim(\mathcal{H})<\infty$	Exp-Regret	$\mathbb{E}_{\mathcal{A}}\left[R_n\right] \leq Ldim(\mathcal{H}) + \sqrt{nLdim(\mathcal{H})\log n}$
Transductive Online Learning (Kakade & Kalai, 2005)	T	$VC(\mathcal{H}) < \infty$	Mistake	$M_n \leq L + n^{3/4}\sqrt{VC(\mathcal{H})\log n}$
Selective Sampling (Cesa-Bianchi et al., 2009)	A/S	Linear	Exp-Regret	$\mathbb{E}\left[R_n\right] \leq \epsilon n_\epsilon + O\left(\frac{1}{\epsilon^{2/\kappa}} + \frac{d}{\epsilon^2}\log n\right)$
EStochAd [*This paper*]	S/A	$VC(\mathcal{H}) < \infty$	HP-Regret	$R_n \leq \sqrt{nVC(\mathcal{H})\log n} + \sqrt{n\log\beta^{-1}}$

Table 1: Performance of algorithms for different classification scenarios. All the bounds are reported up to constant factors. In the setting column, the two letters specify how inputs and labels are generated, where A stands for *adversarial*, S for *stochastic*, and T for *transductive*. In the bound column *HP* stands for high-probability bound and *Exp* stands for bound in expectation. In the perceptron bound M_n is the number of mistakes after n steps, L and D are the cumulative loss and the complexity of any weight matrix. In selective sampling $0 \leq \kappa \leq 1$ is a parameter of the algorithm, n_ϵ is the number of steps with a margin less than ϵ, and the bound holds for any for any $0 < \epsilon < 1$.

Cesa-Bianchi et al. (2009) analyze a learning setting which is complementary to the hybrid setting introduced in this paper. They consider the selective sampling problem in which inputs are arbitrarily generated by an adversary while labels a noisy observations of a linear hypothesis. The main concern in this setting is to limit the number of *queries*, that is the number of times the algorithm asks for the true label corresponding to an input. In particular, they analyze a semi-supervised variant of regularized least squares which approaches the performance of a Bayes optimal classifier with a number of queries sublinear in the time horizon.

From an algorithmic point of view, the use of previous inputs to update the set of hypotheses at the beginning of each epoch resembles the use of unsupervised samples in semi-supervised learning. Similar to the analysis in Kaariainen (2005), we decomposed the regret in a learning performance term, which depends on the actual sequence of labels, and in the approximation of the structure of the inputs marginal distribution term, which just depends on unlabeled instances.

The possibility to convert batch algorithms for the fully stochastic into learning algorithm for the transductive online learning scenario is studied in Kakade & Kalai (2005). In transductive online learning the samples are adversarially generated and all the inputs are known to the learner beforehand. In this scenario, they prove that a batch algorithm can be efficiently translated into an online algorithm with a mistake bound of the order $n^{3/4}\sqrt{d \log n}$ with d the VC-dimension of the hypothesis set. The transductive setting is very similar to the preliminary scenario we described in Section 3.2 in which we assume the sequence of inputs to be known in advance to the learner. In the rest of the paper we showed that in order to move from a transductive setting to a fully online problem and preserve similar results we need to assume the inputs to be drawn independently from some fixed distribution even if this distribution is not known.

Abernethy et al. (2009) compare adversarial on-line learning and statistical learning in online convex optimization. In particular, their analysis reveals that the optimal regret in online convex optimization can be written as the difference between a sum of minimal expected losses and the minimal empirical loss, where samples are generated by an adversarially chosen stochastic process. As a result, it is possible to derive upper and lower bounds for the optimal regret which exhibit several similarities to results for the fully stochastic setting. For instance, they derive an upper bound on the optimal regret in terms of the Rademacher averages of the space of loss functions

induced by the space of functions $\mathcal{F}$ used by the learner.

A direct comparison of the performance of EStochAd with other algorithms for either fully adversarial or fully stochastic settings is difficult because of the different assumptions. Nonetheless, in the following we discuss similarities and differences between EStochAd and other existing algorithms for online prediction. In Table 6, we summarize the main approaches to the classification problem in both stochastic and adversarial settings. Unfortunately not all the bounds are immediately comparable. Some of the regret bounds are in expectation (with respect to either the distribution P or the randomized algorithm $\mathcal{A}$), while others are high-probability bounds. Perceptron performance is stated in terms of mistake bound.

It is interesting to notice that EStochAd incurs exactly the same regret rate as an empirical risk minimization algorithm run online in the fully stochastic case (see Lemma 6). This means that under the assumption that inputs are i.i.d. from a fixed distribution P, the adversarial output does not cause any worsening in the performance with respect to a stochastic output. This result can be explained by the definition of the VC-dimension itself. In fact, while the definition of the VC-dimension requires samples to be generated from a distribution, no assumption is made on the way outputs are generated and any possible sequences of labels is considered. Therefore, it is not surprising that the VC-dimension can be used as a complexity measure for both the case of stochastic and adversarial classification. However, the situation is significantly different in the case of a fully adversarial setting where also inputs can be arbitrarily chosen by an adversary.

Both EStochAd and the Agnostic Online Learning (AOL) algorithm proposed by Ben-David et al. (2009) consider the problem of binary classification with adversarial outputs, an infinite number of hypotheses (experts), and they both build on the exponentially weighted forecaster (Cesa-Bianchi & Lugosi, 2006). On the other hand, the main difference is that while with adversarial inputs it is necessary to consider the Littlestone dimension of $\mathcal{H}$ (Littlestone, 1988), the stochastic assumption on the inputs allows EStochAd to refer to the VC-dimension which is a more natural measure of complexity of the hypothesis space. Moreover, the dependency of the two algorithms on the hypothesis space complexity is different (see Table 6). While AOL has a linear dependency on $Ldim(\mathcal{H})$, in EStochAd the regret grows as $\sqrt{VC(\mathcal{H})}$. Furthermore, as proved by Littlestone (1988), for any hypothesis space $\mathcal{H}$, $VC(\mathcal{H}) \leq Ldim(\mathcal{H})$. In the following we discuss an example showing how in some cases the difference between VC and Littlestone dimension

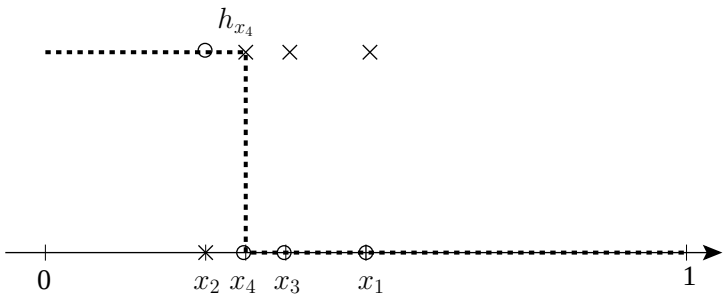

Figure 7: Example of a sequence of inputs and labels such that the adversary can force any learning algorithm to incur a mistake at each round. Circles represent the labels predicted by the learner and crosses the labels revealed by the adversary. h_{x_4} (in dotted-line) is an example of a hypothesis which perfectly classifies all the samples shown so far.

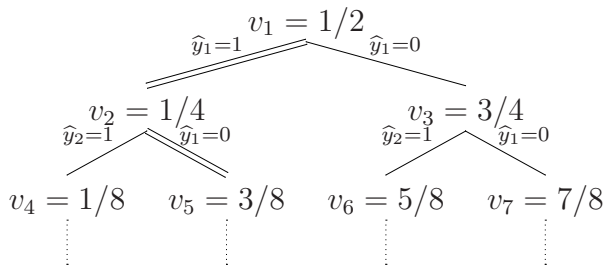

Figure 8: The mistake-tree is defined for any possible sequence of predictions. Double lines correspond to the example depicted in Figure 7.

may be arbitrarily large. Let consider a binary classification problem with $X = [0, 1]$ and a hypothesis space $\mathcal{H}$ containing functions of the form

$$h_\vartheta(x) = \begin{cases} 1 & \text{if } x \geq \vartheta \\ 0 & \text{otherwise,} \end{cases}$$

with $\vartheta \in [0, 1]$.

In the fully adversarial case the regret of AOL is linear in the time horizon (i.e., in the worst case it can make a mistake at each time step). In fact, it can be shown that the Littlestone dimension of $\mathcal{H}$ is infinite. According to Littlestone (1988), the Littlestone dimension is the largest number of mistakes any learning algorithm could incur for any possible sequence of predictions in the realizable case when the adversary is allowed to choose the true label after observing the learner's prediction. Thus, the adversary selects inputs x_t and labels y_t so as to force the learner to make as many

mistakes as possible given the condition that there exists a hypothesis h^* in $\mathcal{H}$ such that $h^*(x_t) = y_t, \ \forall t \leq n$. In order to determine the Littlestone dimension of $\mathcal{H}$ we sketch how to build a shattered mistake-tree of depth n, for any $n > 0$ (see Figure 8). Nodes of the mistake-tree represent the inputs revealed by the adversary depending on the sequence of learner's predictions. Let $v_1 = \frac{1}{2}$ be the root of the mistake-tree, that is the first input x_1 revealed to the algorithm. Next, we label nodes v_2 and v_3 as the middle points of intervals $[0, v_1]$ and $[v_1, 1]$ respectively. The second input shown to the learner depends on the prediction at time $t = 1$. If the prediction is $\widehat{y}_1 = 1$ [4], then the adversary selects a label $y_1 = 0$ and the next input point is set to $x_2 = v_2$. If the algorithm predicts $\widehat{y}_2 = 0$ in x_2, it is still possible to force the algorithm to incur a mistake by setting $y_2 = 1$ without violating the realizability condition. In fact, any hypothesis with $x_2 \leq \vartheta < x_1$ perfectly classifies both y_1 and y_2. The next input x_3 is the middle point of interval $[x_2, x_1]$ and the algorithm is forced to make another mistake. The same process can be repeated at each round by choosing the next input to be the middle point of either the left or the right interval depending on the previous prediction and by revealing a label which is exactly the opposite of the one predicted by the learner. At each step the adversary can force the learner to make a mistake while guaranteeing that it is always possible to find a hypothesis in $\mathcal{H}$ that would make no mistakes (see Figure 7 for the sequence of inputs x_1, x_2, x_3, x_4). As a result, $Ldim(\mathcal{H}) = \infty$ and the AOL has a linear regret. On the other hand, when inputs cannot be arbitrarily chosen by an adversary but are sampled from a fixed distribution EStochAd can achieve a sub-linear regret. In fact, $\mathcal{H}$ could shatter at most one points, the VC-dimension of $\mathcal{H}$ is 1, thus leading a regret for EStochAd of order $O(\sqrt{n \log n})$.

Therefore, even in very simple problems the possibility for the adversary to select the inputs may lead to an arbitrarily bad performance, while drawing inputs from a distribution allows the learner to achieve a sub-linear regret even if outputs are adversarial.

7. Conclusions

In this paper we introduced the hybrid stochastic-adversarial online prediction problem in which inputs are independently and identically generated

[4] The case $\widehat{y}_1 = 0$ is symmetric.

and labels are arbitrarily chosen by an adversary. We devised an epoch-based algorithm for the specific problem of binary classification with full information and analyzed its regret. In particular, we noticed that while the stochastic assumption on inputs allows to use the well-known VC-dimension as a measure of complexity for the hypothesis space, adversarial labels do not cause any worsening in the performance with respect to fully stochastic algorithms. We believe that this analysis, together with its relationship with the results for the fully adversarial case, sheds light on the similarities and differences between batch stochastic learning and adversarial online learning along the line of Kakade & Kalai (2005). Finally, we discussed extensions to multi-label classification, regression, learning from experts and bandits settings with stochastic side information, and approximation of Nash equilibria in games.

In the following we summarize some of the open questions that we plan to investigate in the future.

- **VC-learnability.** The main result of this paper is that any learning setting in which inputs are stochastic is learnable using finite VC-dimension hypothesis spaces independently from the way labels are generated. As noticed in Kakade & Kalai (2005) and Abernethy et al. (2009), strong connections between adversarial online learning and statistical learning can be drawn also in other settings, such as online convex optimization and online transductive learning. On the other hand, the analysis in Ben-David et al. (2009) clearly shows that in the fully adversarial case, the class of learnable problems is smaller than the set of finite VC-dimension spaces. What is the most general online learning setting with the same learnability as fully stochastic problems it is still an open question.

- **Smoothed analysis.** Given an algorithm $\mathcal{A}$, the adversarial online learning setting is a worst-case analysis of the regret over both inputs and labels, while in the hybrid setting introduced in this paper the performance is evaluated according to a worst-case analysis for the labels and average-case for the inputs [5]. Smoothed analysis (Spielman & Teng, 2004) is an alternative to the standard worst-case and average-case analyses in which the smoothed complexity of an algorithm is the

[5] More precisely we provide a high-probability analysis.

maximum over its inputs of the expected performance under slight perturbations of that input. We plan to investigate the use of smoothed-analysis tools to derive a bound explaining both the hybrid and adversarial settings as extremes conditions on the perturbations on the inputs.

- **Efficient algorithm.** The analysis in Section 4 shows that, although polynomial in the time horizon, EStochAd has an exponential complexity w.r.t. the VC-dimension d. This dependency makes the algorithm inefficient both in terms of time and space complexity when the hypothesis space $\mathcal{H}$ has a high VC-dimension. Whether it is possible to obtain an efficient algorithm with the same regret is still an open question. We conjecture that a more numerically efficient algorithm may come at the cost of a worsening of the regret as in the transductive setting in Kakade & Kalai (2005).

Acknowledgments. This work has been supported by French National Research Agency (ANR) through COSINUS program (project EXPLO-RA nANR-08-COSI-004).

Appendix A. Online Empirical Risk Minimizer

We report the lemma stating the regret of an empirical risk minimizer run online in the fully stochastic setting.

Lemma 6. *Let $\{x_t, y_t\}_{t=1}^n \overset{iid}{\sim} P$ be a sequence of i.i.d. input-label pairs drawn from a distribution P and $\mathcal{H}$ be a hypothesis space with finite VC-dimension $d = VC(\mathcal{H}) < \infty$. At each round t the learner returns the hypothesis minimizing the cumulative loss*

$$h_t = \arg\min_{h \in \mathcal{H}} \sum_{s=1}^{t-1} \ell(h_s(x_s), y_s),$$

For any $n > 0$, the learner achieves a regret

$$\mathbb{E}\left[R_n\right] \leq 2\sqrt{2}\sqrt{nd \log \frac{2en}{d}} + 2\sqrt{2}\sqrt{n \log \frac{2n}{\beta}}$$

with probability $1 - \beta$.

Proof. Let

$$\bar{\ell}(h) = \mathbb{E}_{(x,y)\sim P}\left[\ell(h(x), y)\right]; \quad \hat{\ell}_t(h) = \frac{1}{t}\sum_{s=1}^{t} \ell(h(x_s), y_s).$$

Let h^* be the expected loss minimizer, that is $h^* = \arg\inf_{h \in \mathcal{H}} \bar{\ell}(h)$. We prove the following sequence of inequalities.

$$\begin{aligned}
\mathbb{E}\left[R_n\right] &= \sum_{t=1}^{n} \left(\bar{\ell}(h_t) - \bar{\ell}(h^*)\right) \\
&\overset{(a)}{\leq} \sum_{t=1}^{n} \left(\bar{\ell}(h_t) - \hat{\ell}_{t-1}(h_t) + \hat{\ell}_{t-1}(h^*) - \bar{\ell}(h^*)\right) \\
&\overset{(b)}{\leq} \sum_{t=1}^{n} 2 \sup_{h \in \mathcal{H}} \left|\bar{\ell}(h) - \hat{\ell}_{t-1}(h)\right| \\
&\overset{(c)}{\leq} 2\sqrt{2} \sum_{t=1}^{n} \left(\sqrt{\frac{d}{t-1} \log \frac{2e(t-1)}{d}} + \sqrt{\frac{1}{t-1} \log \frac{2}{\beta'}}\right) \quad \text{w.p.} 1 - n\beta' \\
&\overset{(d)}{\leq} 2\sqrt{2}\sqrt{nd \log \frac{2en}{d}} + 2\sqrt{2}\sqrt{n \log \frac{2}{\beta'}}.
\end{aligned}$$

(a) By definition of h_t it is the hypothesis minimizing the empirical loss, thus $\hat{\ell}_{t-1}(h^*) - \hat{\ell}_{t-1}(h_t) \geq 0$.
(b) We take the supremum over all the hypotheses in $\mathcal{H}$.
(c) An application of a VC-bound (Bousquet et al., 2004).
(d) Result of the sum over n rounds.

The statement follows by setting $\beta = n\beta'$.

□

Appendix B. Functional Azuma's Inequality

In this section we prove an extension of the Azuma's inequality to a hypothesis space $\mathcal{H}$. First we recall the definition of martingale difference sequence and the Hoeffding-Azuma's inequality.

Definition 3. *A sequence of random variables $z_1, z_2, \ldots$ is a martingale difference sequence with respect to a sequence of random variables $x_1, x_2, \ldots$ if*

$$\mathbb{E}\left[z_{t+1} | x_1, \ldots, x_t\right] = 0,$$

with probability 1 for any $t > 0$.

Proposition 1. *Let $z_1, z_2, \ldots$ be a martingale difference sequence with respect to a sequence $x_1, x_2, \ldots$. Assume furthermore that there exists a sequence of nonnegative constants $c_1, c_2, \ldots$ such that $|z_t| \leq c_t$ for any $t > 0$. Then for any $\epsilon > 0$ and n*

$$\mathbb{P}\left[\sum_{t=1}^{n} z_t \geq \epsilon\right] \leq \exp\left(-\frac{\epsilon^2}{2\sum_{t=1}^{n} c_t^2}\right).$$

Now we extend the previous theorem to the functional case on a space of binary functions.

Lemma 7. *Let $\mathcal{H}$ be a space of functions $h : \mathcal{X} \to \{0, 1\}$ with finite VC-dimension $VC(\mathcal{H}) = d < \infty$. Assume that $h(x_1), \ldots, h(x_n)$ is a martingale difference sequence with respect to $x_1, \ldots, x_n$ for any $h \in \mathcal{H}$. Then for any $\epsilon > 0$ and n*

$$\mathbb{P}\left[\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{t=1}^{n} h(x_t) \geq \epsilon\right] \leq \left(\frac{en}{d}\right)^d \exp\left(-\frac{n\epsilon^2}{2}\right). \tag{B.1}$$

Proof. The proof is a straightforward application of a union bound on the Azuma's inequality in Proposition 1. Let $\mathcal{H}_{x_1,\ldots,x_n}$ be the space of vectors obtained by evaluating all the functions $h \in \mathcal{H}$ on points $x_1, \ldots, x_n$. By definition of VC-dimension of $\mathcal{H}$ the cardinality of $\mathcal{H}_{x_1,\ldots,x_n}$ is bounded by $\left(\frac{en}{d}\right)^d$ (Sauer, 1972). Thus, we obtain

$$\mathbb{P}\left[\sup_{h\in\mathcal{H}} \frac{1}{n}\sum_{t=1}^{n} h(x_t) \geq \epsilon\right] \leq \mathbb{P}\left[\sup_{h\in\mathcal{H}_{x_1,\ldots,x_n}} \frac{1}{n}\sum_{t=1}^{n} h(x_t) \geq \epsilon\right] \tag{B.2}$$

$$\leq \left(\frac{en}{d}\right)^d \mathbb{P}\left[\sum_{t=1}^{n} h(x_t) \geq n\epsilon\right] \tag{B.3}$$

$$\leq \left(\frac{en}{d}\right)^d \exp\left(-\frac{n\epsilon^2}{2}\right), \tag{B.4}$$

where in the last step we used the Azuma's inequality in Proposition 1. □

Lemma 8. *Let $\mathcal{F}$ be a space of functions $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ with finite VC-dimension $VC(\mathcal{F}) = d < \infty$. Let $x_1, \ldots, x_n$ be a sequence of i.i.d. samples from a distribution P and $\bar{f}(y) = \mathbb{E}_{x\sim P}[f(x,y)]$. Assume that the sequence $y_1, \ldots, y_n$ is such that $\left(f(x_1,y_1) - \bar{f}(y_1)\right), \ldots, \left(f(x_t,y_t) - \bar{f}(y_t)\right)$ is a martingale difference sequence with respect to $x_1, \ldots, x_n, y_1, \ldots, y_n$ for any $f \in \mathcal{F}$. Then for any $\epsilon > 0$ and n*

$$\mathbb{P}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{t=1}^{n} \left(\bar{f}(y_t) - f(x_t,y_t)\right) \geq \epsilon\right] \leq 2\left(\frac{en}{d}\right)^d \exp\left(-\frac{n\epsilon^2}{2}\right). \tag{B.5}$$

Proof. We divide the proof in two steps.

Step (i): Symmetrization

Let $x'_1, \ldots, x'_n$ be a sequence of ghost samples i.i.d. from P. We use symmetrization to replace the average of the expectation $\bar{f}(y_t)$ with an empirical version over the ghost sample. Let $f^* \in \mathcal{F}$ a function such that

$$\frac{1}{n}\sum_{t=1}^{n} \bar{f}^*(y_t) - f^*(x_t,y_t) \geq \epsilon. \tag{B.6}$$

Notice that f^* is a random variable depending of samples $x_1, \ldots, x_n, y_1, \ldots, y_n$. The following sequence of inequalities leads to the symmetrization result:

$$
\begin{aligned}
&\mathbb{P}\left[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{t=1}^{n}\big(f(x_t',y_t)-f(x_t,y_t)\big)\geq\epsilon\right]\overset{\text{(a)}}{\geq}\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(f^*(x_t',y_t)-f^*(x_t,y_t)\big)\geq\epsilon\right]\\
&\overset{\text{(b)}}{\geq}\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t,y_t)\big)\geq\epsilon\wedge\frac{1}{n}\sum_{t=1}^{n}\big(f^*(x_t',y_t)-\bar{f}^*(y_t))\big)\geq-\frac{\epsilon}{2}\right]\\
&\overset{\text{(c)}}{=}\mathbb{E}\left[\mathbb{I}\left\{\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t,y_t)\big)\geq\epsilon\right\}\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t',y_t))\big)\leq\frac{\epsilon}{2}\middle|x_1^ny_1^n\right]\right]\\
&\overset{\text{(d)}}{\geq}\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t,y_t)\big)\geq\epsilon\right]\frac{1}{2}.
\end{aligned}
$$

(a) We restrict the space of functions from $\mathcal{F}$ to the set of functions satisfying condition (B.6).
(b) We introduce the ghost sample.
(c) We write the joint probability as the expectation of the first event times the probability of the second even conditioned on the original sample $x_1,\ldots,x_n,y_1,\ldots,y_n$.
(d) The conditional probability can be lower-bounded as follows

$$
\begin{aligned}
&\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t',y_t))\big)\geq\frac{\epsilon}{2}\middle|x_1^ny_1^n\right]\\
&=\mathbb{P}\left[\mathbb{E}\left[\sum_{t=1}^{n}f^*(x_t',y_t)\right]-\sum_{t=1}^{n}f^*(x_t',y_t)\geq\frac{n\epsilon}{2}\middle|x_1^ny_1^n\right]\\
&\leq\frac{4}{n^2\epsilon^2}\mathrm{Var}\left[\sum_{t=1}^{n}f^*(x_t',y_t)\middle|x_1^ny_1^n\right]\\
&\leq\frac{4}{n^2\epsilon^2}\frac{1}{4}n=\frac{1}{n\epsilon^2}\leq\frac{1}{2},
\end{aligned}
$$

where we used Chebyshev's inequality and the condition $n\epsilon^2>2$. Thus we obtain

$$
\mathbb{P}\left[\frac{1}{n}\sum_{t=1}^{n}\big(\bar{f}^*(y_t)-f^*(x_t,y_t)\big)\geq\epsilon\right]\leq2\mathbb{P}\left[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{t=1}^{n}\left(f(x_t',y_t)-f(x_t,y_t)\right)\geq\epsilon\right]
$$

Step (ii): Azuma's inequality

We now use the functional Azuma's inequality in Lemma 7 to bound the last term. In fact, it easy to notice that

$$\big(f(x_t', y_t) - \bar{f}(y_t)\big) - \big(f(x_t, y_t) - \bar{f}(y_t)\big) \tag{B.7}$$

is a martingale difference sequence ($\big(f(x_t, y_t) - \bar{f}(y_t)\big)$ is martingale difference sequence by assumption). Thus, we obtain

$$\mathbb{P}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{t=1}^{n} \left(f(x_t', y_t) - f(x_t, y_t)\right) \geq \epsilon\right] \leq \left(\frac{en}{d}\right)^d \exp\left(-\frac{n\epsilon^2}{2}\right) \tag{B.8}$$

The final statement follows by putting together the two steps.

□

References

Abernethy, J., Agarwal, A., Bartlett, P. L., & Rakhlin, A. (2009). A stochastic view of optimal regret through minimax duality. In *Proceedings of the 22nd Annual Conference on Learning Theory (COLT-09)*. Montreal, Canada.

Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R. E. (2003). The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, *32*, 48–77.

Ben-David, S., Cesa-Bianchi, N., Haussler, D., & Long, P. M. (1995). Characterizations of learnability for classes of {0...n}-valued functions. *Journal of Computer and System Sciences*, *50*, 74–86.

Ben-David, S., Pal, D., & Shalev-Shwartz, S. (2009). Agnostic online learning. In *Proceedings of the 22nd Annual Conference on Learning Theory (COLT-09)*. Montreal, Canada.

Bousquet, O., Boucheron, S., & Lugosi, G. (2004). Introduction to statistical learning theory. *Advanced Lectures on Machine Learning Lecture Notes in Artificial Intelligence*, *3176*, 169–207.

Cesa-Bianchi, N., Freund, Y., Haussler, D., Helmbold, D. P., Shapire, R., & Warmuth, M. (1997). How to use expert advice. *Journal of the ACM*, *44*, 427–485.

Cesa-Bianchi, N., Gentile, C., & Orabona, F. (2009). Robust bounds for classification via selective sampling. In *Proceedings of the 26th International Conference on Machine Learning (ICML-09)* (pp. 121–128). Montreal, Canada.

Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, Learning, and Games*. Cambridge University Press.

Crammer, K., & Singer, Y. (2003). Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research*, *3*, 951–991.

Fink, M., Shalev-Shwartz, S., Singer, Y., & Ullman, S. (2006). Online multiclass learning by interclass hypothesis sharing. In *Proceedings of the 23rd international Conference on Machine Learning (ICML-06)* (pp. 313–320). Pittsburgh, Pennsylvania.

Haussler, D. (1992). Decision theoretic generalizations of the pac model for neural net and other learning applications. *Inf. Comput.*, *100*, 78–150.

Kaariainen, M. (2005). Generalization error bounds using unlabeled data. In *Proceedings of the 18th Annual Conference on Learning Theory (COLT-05)* (pp. 127–142). Bertinoro, Italy.

Kakade, S. M., & Kalai, A. (2005). From batch to transductive online learning. In *Advances in Neural Information Processing Systems (NIPS-05)*. Vancouver, Canada.

Kakade, S. M., Shalev-Shwartz, S., & Tewari, A. (2008). Efficient bandit algorithms for online multiclass prediction. In *Proceedings of the 25th International Conference on Machine Learning (ICML-08)* (pp. 440–447). Helsinki, Finland.

Langford, J., & Zhang, T. (2007). The epoch greedy algorithm for contextual multi-armed bandits. In *Advances in Neural Information Processing Systems (NIPS-07)*. Vancouver, Canada.

Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, *2*, 285–318.

Littlestone, N., & Warmuth, M. (1994). The weighted majority algorithm. *Information and Computation*, *108*, 212–261.

Natarajan, B. K. (1989). On larning sets and functions. *Machine Learning*, *4*, 67–97.

Pollard, D. (1984). *Convergence of Stochastic Processes*. Springer Verlag, New York.

Rosenblatt, F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*, 386–408.

Ryabko, D. (2006). Pattern recognition for conditionally independent data. *Journal of Machine Learning Research*, *7*, 645–664.

Sauer, N. (1972). On the densities of families of sets. *Journal of Combinatorial Theory (A)*, *13*, 145–147.

Spielman, D. A., & Teng, S.-H. (2004). Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. ACM*, *51*, 385–463.

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

Vovk, V. (1998). A game of prediction with expert advice. *Journal of Computer and System Sciences*, *56*, 153–173.

Weston, J., & Watkins, C. (1999). Support vector machines for multi-class pattern recognition. In *Proceedings of the Seventh European Symposium on Artificial Neural Networks*.