Journal of Machine Learning Research 17 (2016) 1-21 Submitted 11/13; Revised 10/15; Published 4/16

A Closer Look at Adaptive Regret

Dmitry Adamskiy ADAMSKIY@CS.RHUL.AC.UK
Computer Learning Research Centre and Department of Computer Science,
Royal Holloway, University of London, Egham, Surrey, TW20 0EX, UK

Wouter M. Koolen WMKOOLEN@CWI.NL
Centrum Wiskunde & Informatica
Science Park 123, 1098XG Amsterdam, The Netherlands

Alexey Chernov A.CHERNOV@BRIGHTON.AC.UK
School of Computing, Engineering and Mathematics, University of Brighton
Moulsecoomb, Brighton, BN2 4GJ, UK

Vladimir Vovk VOVK@CS.RHUL.AC.UK
Computer Learning Research Centre and Department of Computer Science,
Royal Holloway, University of London, Egham, Surrey, TW20 0EX, UK

Editor: Manfred Warmuth

Abstract

For the prediction with expert advice setting, we consider methods to construct algorithms that have low adaptive regret. The adaptive regret of an algorithm on a time interval $[t_1, t_2]$ is the loss of the algorithm minus the loss of the best expert over that interval. Adaptive regret measures how well the algorithm approximates the best expert locally, and so is different from, although closely related to, both the classical regret, measured over an initial time interval $[1, t]$, and the tracking regret, where the algorithm is compared to a good sequence of experts over $[1, t]$.

We investigate two existing intuitive methods for deriving algorithms with low adaptive regret, one based on specialist experts and the other based on restarts. Quite surprisingly, we show that both methods lead to the same algorithm, namely Fixed Share, which is known for its tracking regret. We provide a thorough analysis of the adaptive regret of Fixed Share. We obtain the exact worst-case adaptive regret for Fixed Share, from which the classical tracking bounds follow. We prove that Fixed Share is optimal for adaptive regret: the worst-case adaptive regret of any algorithm is at least that of an instance of Fixed Share.

Keywords: online learning, adaptive regret, Fixed Share, specialist experts

1. Introduction

This paper deals with the prediction with expert advice setting. Nature generates outcomes step by step. At every step Learner tries to predict the outcome. Then the actual outcome is revealed and the quality of Learner's prediction is measured by a loss function.

No assumptions are made about the nature of the data. Instead, at every step Learner is presented with the predictions of a pool of experts and he may base his predictions on these. The goal of Learner in the classical setting is to guarantee small regret, that is, to suffer cumulative loss that is not much larger than that of the best (in hindsight) expert

from the pool. Several classical algorithms exist for this task, including the Aggregating Algorithm (Vovk, 1990) and the Exponentially Weighted Forecaster (Cesa-Bianchi and Lugosi, 2006). In the standard logarithmic loss game the regret incurred by those algorithms when competing with N experts is at most $\ln N$, independent of the number of steps.

A common extension of the framework takes into account the fact that the best expert could change with time. In this case we may be interested in competing with the best *sequence* of experts from the pool. Known algorithms for this task include Fixed Share (Herbster and Warmuth, 1998) and Mixing Past Posteriors (Bousquet and Warmuth, 2002).

In this paper we focus on the related task of obtaining small *adaptive* regret, a notion first considered by Littlestone and Warmuth (1994) and later studied by Hazan and Seshadhri (2009). The adaptive regret of an algorithm on a time interval $[t_1, t_2]$ is the loss that the algorithm accumulates there, minus the loss of the best expert for that interval:

$$R_{[t_1,t_2]} \coloneqq L_{[t_1,t_2]} - \min_n L^n_{[t_1,t_2]} \,. \tag{1}$$

The goal is now to ensure small adaptive regret on all intervals simultaneously. Note that adaptive regret was defined by Hazan and Seshadhri (2009) with a maximum over intervals, but we need the fine-grained dependence on the endpoint times to be able to prove matching upper and lower bounds.

Our results. The contribution of our paper is threefold.

1. We study two constructions to get adaptive regret algorithms from existing classical regret algorithms. The first one is a simple construction proposed by Freund et al. (1997) and slightly generalised by Chernov and Vovk (2009) that involves so called specialists (sleeping experts), and the second one uses restarts, as proposed by Hazan and Seshadhri (2009). Although conceptually dissimilar, we show that both constructions yield the Fixed Share algorithm with a time-varying switching rate.

2. We compute the exact worst-case adaptive regret of Fixed Share. We re-derive the tracking regret bounds from these adaptive regret bounds, showing that the latter are in fact more fundamental.

3. We show that Fixed Share is the optimal algorithm for adaptive regret, in the sense that the worst-case adaptive regret of any candidate algorithm is at least that of an instance of Fixed Share.

Here is a sneak preview of the adaptive bounds we obtain, presented in a slightly relaxed form for simplicity. The refined statement can be found in Theorem 4 below. In the logarithmic loss game for each of the following adaptive regret bounds there is an algorithm satisfying it, simultaneously for all intervals $[t_1, t_2]$:

$$\ln N + \ln t_2 \,, \tag{2a}$$

$$\ln N + \ln t_1 + \ln\ln t_2 + 2 \,, \tag{2b}$$

$$\ln N + 2\ln t_1 + 1 \,, \tag{2c}$$

where $\ln\ln 1$ is interpreted as 0.

Protocol 1 Mix-loss prediction

for $t = 1, 2, \ldots$ **do**
  Learner announces probability vector $\boldsymbol{u}_t \in \triangle_N$
  Reality announces vector $\boldsymbol{\ell}_t \in (-\infty, \infty]^N$ of expert losses
  Learner suffers loss $\ell_t := -\ln \sum_n u_t^n e^{-\ell_t^n}$
end for

Outline. The structure of the paper is as follows. In Section 2 we give the description of the protocol and review standard algorithms. In Section 3 we study two intuitive ways of obtaining adaptive regret algorithms from classical algorithms. We show that curiously both resulting algorithms turn out to be Fixed Share. In Section 4 we study in detail the adaptive regret of Fixed Share and establish its optimality.

2. Setup

We phrase our results in the setting defined in Protocol 1, which, for lack of a standard name, we call *mix loss*. We choose this fundamental setting because it is universal, in the sense that many other common settings reduce to it. For example probability forecasting, sequential investment and data compression are straightforward instances (Cesa-Bianchi and Lugosi, 2006).[1] In addition, mix loss is the baseline for the wider class of *mixable loss functions*, which includes e.g. square loss (Vovk, 2001). Classical (entire $[1, t]$ interval) regret upper bounds transfer from mix loss to mixable losses almost by definition, and the same reasoning extends to adaptive regret bounds (see Appendix A). In addition, mix-loss methods and upper bounds carry over in a modular way (via the individual-sequence versions of Hoeffding-type bounds, e.g. by Cesa-Bianchi et al. 2012) to non-mixable games, which include the Hedge setting (Freund and Schapire, 1997) and Online Convex Optimisation (Zinkevich, 2003). The number N of experts is fixed throughout the paper.

Let us review the specialisation for our setup of two standard algorithms.[2] The *Aggregating Algorithm*, or AA, by Vovk (1998) predicts[3] in trial t with

$$u_t^n := \frac{e^{-\sum_{s<t} \ell_s^n}}{\sum_j e^{-\sum_{s<t} \ell_s^j}}, \tag{3a}$$

which we may also maintain incrementally using the update rule

$$u_{t+1}^n = \frac{u_t^n e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}}. \tag{3b}$$

1. Namely, if ℓ_t^n contains the negative log returns of stock n over trading round t, the mix loss is the negative log return of the portfolio $\boldsymbol{u}_t$. Similarly, if ℓ_t^n contains the negative log likelihoods that probability model n assign to the t-th outcome, then the mix loss is the negative log likelihood of the model average $\boldsymbol{u}_t$.
2. The algorithms we review maintain weights on experts, and originally come with a strategy for subsequently issuing predictions by aggregating the experts' predictions. For mix loss the predictions are abstracted away and an algorithm is evaluated just by its weights.
3. The AA can be parametrised by a prior distribution. As we only need the uniform prior in this section we specialised to that case immediately. The same holds for Fixed Share below.

For this algorithm the classical regret bound states that for each expert n

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_t^n \;\leq\; \ln N \tag{4}$$

(here and below $\infty - \infty$ is interpreted as 0; i.e., Learner never feels regret w.r.t. an expert who suffers infinite loss). Note that the AA is minimax for classical mix-loss regret since regret $\geq \ln N$ can be inflicted on any algorithm already in the first round.

The second algorithm, *Fixed Share* by Herbster and Warmuth (1998), requires a sequence of switching rates $\alpha_2, \alpha_3, \ldots$. Intuitively, α_t is the probability of a switch to a different expert in the sequence of "best-at-the-step" experts between trials $t-1$ and t. Like the AA, FS starts with uniform weights $u_1^n = 1/N$. The weights are now updated as

$$u_{t+1}^n \;\coloneqq\; \frac{\alpha_{t+1}}{N-1} + \left(1 - \frac{N}{N-1}\alpha_{t+1}\right) \frac{u_t^n e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}}\,. \tag{5}$$

The intuition behind this expression is that if an expert's normalized weight is w and each expert redistributes a fraction α of his weight uniformly to the other experts, his resulting weight will become $\frac{\alpha}{N-1}(1-w) + (1-\alpha)w = \frac{\alpha}{N-1} + (1 - \frac{N}{N-1}\alpha)w$. We see that the AA is the special case when all α_t are 0 (on the other hand, Fixed Share is a special case of the AA for a certain infinite set of experts as mentioned by Vovk 1999). The tracking regret bound by Herbster and Warmuth (1998) for Fixed Share with constant $\alpha_t = \alpha$ switching rate states that for any reference sequence $n_1, \ldots, n_T$ of experts with m blocks (and hence $m-1$ switches)

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_t^{n_t} \;\leq\; \ln N + (m-1)\ln(N-1) - (m-1)\ln\alpha - (T-m)\ln(1-\alpha). \tag{6}$$

We will see later (Lemma 11 below) that the interesting values of α_t are in the range $[0, \frac{N-1}{N}]$. Intuitively, a value $\alpha_t > \frac{N-1}{N}$ corresponds to assigning larger weights to poor experts, and this always hurts the worst-case adaptive regret (in the borderline case $\alpha = \frac{N-1}{N}$, (5) becomes $u_{t+1}^n \coloneqq \frac{1}{N}$). We will also set

$$\alpha_1 \coloneqq \frac{N-1}{N}; \tag{7}$$

since the algorithm only involves $\alpha_2, \alpha_3, \ldots$ this causes no harm (but simplifies some formulas). Having introduced the standard classical and tracking regret algorithms, we now turn to adaptive regret.

3. Intuitive Algorithms with Low Adaptive Regret

Two methods have been proposed in the literature that can be used to obtain adaptive regret bounds: specialists (sleeping experts) (Freund et al., 1997) and restarts (Hazan and Seshadhri, 2009). We discuss both and show that each of them yields Fixed Share with a particular choice of time-dependent switching rate α_t.

3.1 Specialist Experts

To discuss the first method we need a simple extension of the mix-loss prediction protocol to the case of *specialist experts*, who are absent at some steps ("are asleep"). At the beginning of each round t the subset $A_t \subseteq \{1,\dots,N\}$ of experts who are awake is revealed, and the other experts are said to be asleep. The algorithm is required to assign probabilities only to the experts who are awake, and its loss is now defined by the formula $\ell_t := -\ln \sum_{n\in A_t} u_t^n e^{-\ell_t^n}$. The *specialist AA* is the extension of the AA to specialists. Like the AA it maintains weights on all experts, starting from some prior $\boldsymbol{u}_1$. Each round, it predicts by conditioning the current weights $\boldsymbol{u}_t$ on the set A_t of experts who are awake, thus assigning to such an expert n weight $u_t^n / \sum_{j\in A_t} u_t^j$. After observing the losses the weight of each expert who is awake, $n \in A_t$, is updated multiplicatively as $u_{t+1}^n := u_t^n e^{\ell_t - \ell_t^n}$ and the weight of each sleeping expert $n \notin A_t$ stays put at $u_{t+1}^n := u_t^n$.

The AA and specialist AA are related in a useful manner: Chernov and Vovk (2009) obtain the specialist AA from the AA by imagining that all sleeping experts suffer the same loss as the algorithm. This observation immediately leads to a relativised analogue of regret bound (4). The specialist AA guarantees, for each specialist n,

$$\sum_{t\le T: n\in A_t} \ell_t - \sum_{t\le T: n\in A_t} \ell_t^j \;\le\; \ln N. \tag{8}$$

The loss of expert n is defined only during the rounds when he is awake. This bound tells us that the cumulative loss of the specialist AA incurred during those rounds does not exceed the cumulative loss of expert n by much.

We now turn to obtaining adaptive regret bounds for the vanilla expert setting by running the specialist AA on imaginary (virtual) sleeping experts of our own design.

3.1.1 Specialist Experts

One way of getting an adaptive algorithm is the following. We create a pool of virtual experts. For each real expert n and time t, we include a virtual expert that sleeps during the first $t-1$ trials, and subsequently predicts as expert n from trial t onward. The specialist regret (in the sense of 8) w.r.t. this virtual expert on $[1,T]$ is the same as the adaptive regret w.r.t. the real expert n on $[t,T]$. The natural idea is to feed all those virtual experts into the existing algorithm capable of obtaining good classical regret, the specialist AA. For fixed t_2, the uniform prior on wake-up time $t_1 \le t_2$ and expert n this would lead to adaptive regret $\ln(Nt_2)$. It turns out that the same holds even without knowledge of t_2.

At first glance, it is very inefficient, even in the case of a finite horizon T, to maintain weights of TN specialists. However, we do not need to, since we may merge the weights of all specialists who are awake and associated to the same real expert, resulting in Algorithm 1. To verify that this algorithm is correct, denote this merged (unnormalised) weight in trial t by v_t^n for each real expert n. The merged (unnormalised) weight v_{t+1}^n of this real expert n in the next trial $t+1$ consists of the prior weight, denoted $p(t+1)$, of the newly awaken virtual specialist plus v_t^n, the sum of the weights of the previously awaken specialists, each multiplied by the same factor $e^{\ell_t - \ell_t^n}$ (as they were all awake). Thus we can update the sum directly, and this is reflected by our update rule.

Algorithm 1 Adaptive Aggregating Algorithm

Input: Prior nonnegative weights $p(t)$, $t = 1, 2, \ldots$, with $p(1) > 0$
$v_1^n := p(1)$, $n = 1, \ldots, N$
for $t = 1, 2, \ldots$ **do**
Play weights $u_t^n := \frac{v_t^n}{\sum_{j=1}^N v_t^j}$
Read the experts losses ℓ_t^n, $n = 1, \ldots, N$
Set $v_{t+1}^n := p(t+1) + v_t^n \frac{e^{-\ell_t^n}}{\sum_{j=1}^N u_t^j e^{-\ell_t^j}}$, $n = 1, \ldots, N$
end for

Note that for simplicity, we have taken the (unnormalised) priors on experts and wake-up times independent, i.e.

$$p^{(n,t)} \;=\; p(t)\,.$$

(There is no need for the prior weights $p^{(n,t)}$ to normalise, as the predictions are normalised explicitly.)

Now we will see that Algorithm 1 turns out to be Fixed Share with variable switching rate. In the rest of this section we derive this. Let $P(t) = \sum_{s=1}^t p(s)$.

Fact 1 *The update step of Algorithm 1 preserves the following: for all $t \geq 1$*

$$\sum_n v_t^n = \sum_n \sum_{s \leq t} p(s) = NP(t)\,.$$

Proof This follows immediately from expanding the one-step update rule:

$$\begin{aligned}
\sum_n v_{t+1}^n \;&=\; \sum_n p(t+1) + \sum_n v_t^n \frac{e^{-\ell_t^n}}{\sum_k u_t^k e^{-\ell_t^k}} \\
&=\; \sum_n p(t+1) + \sum_n v_t^n \frac{e^{-\ell_t^n}}{\sum_k \frac{v_t^k}{\sum_n v_t^n} e^{-\ell_t^k}} \\
&=\; Np(t+1) + \sum_n v_t^n \;\overset{\text{Induction}}{=}\; NP(t+1)\,. \qquad \blacksquare
\end{aligned}$$

We now show that Algorithm 1 can be seen as Fixed Share (and vice versa).

Lemma 2 *Suppose the probabilities $\alpha_t \in [0, \frac{N-1}{N}]$ of a Fixed Share switch before trial t and the prior weights $p(t)$ of a specialist waking up in trial t in Algorithm 1 satisfy*

$$p(t) \;=\; \frac{\frac{N}{N-1}\alpha_t}{\prod_{s=2}^t (1 - \frac{N}{N-1}\alpha_s)}$$

(where we use the convention 7) or, equivalently,

$$\alpha_t \;=\; \frac{N-1}{N} \frac{p(t)}{\sum_{s=1}^t p(s)}\,.$$

Then the two algorithms output identical predictions.

Proof Let us rewrite the update step of Algorithm 1 for the normalised weights.

$$\begin{aligned} u_{t+1}^n &= \frac{v_{t+1}^n}{\sum_j v_{t+1}^j} = \frac{p(t+1)}{NP(t+1)} + \frac{1}{NP(t+1)} v_t^n \frac{e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}} \\ &= \frac{p(t+1)}{NP(t+1)} + \frac{1}{NP(t+1)} NP(t) u_t^n \frac{e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}} \\ &= \frac{\alpha_{t+1}}{N-1} + \frac{P(t+1) - p(t+1)}{P(t+1)} u_t^n \frac{e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}} \\ &= \frac{\alpha_{t+1}}{N-1} + \left(1 - \frac{N}{N-1}\alpha_{t+1}\right) u_t^n \frac{e^{-\ell_t^n}}{\sum_j u_t^j e^{-\ell_t^j}}. \end{aligned}$$

We see that the weight update is the update of the Fixed Share algorithm with variable switching rate α_t. ■

The idea to use specialist experts for obtaining adaptive bounds was introduced by Freund et al. (1997). There a virtual specialist is created for every interval $[t_1, t_2]$ which leads to redundancy and suboptimal bounds. Their adaptive regret bounds include a term which exceeds $2 \ln t_2$ whereas our bounds (2) have at most a single $\ln t_2$.

3.2 Restarts

A second intuitive method to obtain adaptive regret bounds, called Follow the Leading History (FLH), was introduced by Hazan and Seshadhri (2007, 2009).[4] One starts with a base algorithm that ensures low classical regret. FLH then obtains low adaptive regret by restarting a copy of this base algorithm at each trial, and aggregating the predictions of these copies. To get low adaptive regret w.r.t. N experts[5], it is natural to take the AA as the base algorithm. We now show that FLH with this choice equals Fixed Share with switching rate $\alpha_t = \frac{N-1}{Nt}$.

For each n, s and $t \geq s$, let $p_t^{n|s}$ denote the weight allocated to expert n by the copy of the AA started at time s. By definition $p_s^{n|s} = 1/N$, and these weights evolve according to (3b). We denote by p_t^s the weight allocated by FLH in trial $t \geq s$ to the copy of the AA started at time s. Hazan and Seshadhri (2009) define these weights as follows. Initially $p_1^1 = 1$ and subsequently

$$p_{t+1}^{t+1} = \frac{1}{t+1} \quad \text{and} \quad p_{t+1}^s = \left(1 - p_{t+1}^{t+1}\right) \frac{p_t^s e^{-\left(-\ln \sum_n p_t^{n|s} e^{-\ell_t^n}\right)}}{\sum_{r=1}^t p_t^r e^{-\left(-\ln \sum_n p_t^{n|r} e^{-\ell_t^n}\right)}} \quad \text{for all } 1 \leq s \leq t.$$

4. Here we present the version of Follow the Leading History with the best performancee guarantee. This version is called FLH1 by Hazan and Seshadhri (2007). It can be recovered from FLH by Hazan and Seshadhri (2009) by omitting the pruning step, which considerably improves computational efficiency at the cost of predictive performance. Alghough such tradeoffs are not the focus of our paper, they are of great practical significance. We refer to György et al. (2012) for an in-depth analysis.
5. More broadly, for any exp-concave loss function FLH can upgrade classic regret bounds for any baseline algorithm to their adaptive analogues, resulting in efficient adaptive regret algorithms for continuous online optimisation.

We now show that this construction is a reparametrisation of Fixed Share. In fact, this is true for any choice of the restart probabilities p_t^t.

Lemma 3 *For mix loss, FLH with the AA as the base algorithm issues the same predictions as Fixed Share with learning rate* $\alpha_t = \frac{N-1}{N} p_t^t$.

Proof We prove by induction on t that the FS and FLH weights coincide:

$$u_t^n \;=\; \sum_{s=1}^{t} p_t^{n|s} p_t^s.$$

The base case $t = 1$ is obvious. For the induction step we expand

$$\begin{aligned}
\sum_{s=1}^{t+1} p_{t+1}^{n|s} p_{t+1}^s \;&=\; \sum_{s=1}^{t} p_{t+1}^{n|s} p_{t+1}^s + p_{t+1}^{t+1}/N \\
&=\; \left(1 - p_{t+1}^{t+1}\right) \sum_{s=1}^{t} \left(\frac{p_t^{n|s} e^{-\ell_t^n}}{\sum_n p_t^{n|s} e^{-\ell_t^n}} \frac{p_t^s \left(\sum_n p_t^{n|s} e^{-\ell_t^n}\right)}{\sum_{r=1}^{t} p_t^r \left(\sum_n p_t^{n|r} e^{-\ell_t^n}\right)} \right) + \frac{1}{N} p_{t+1}^{t+1} \\
&=\; \left(1 - p_{t+1}^{t+1}\right) \frac{\sum_{s=1}^{t} p_t^s p_t^{n|s} e^{-\ell_t^n}}{\sum_{r=1}^{t} \sum_n p_t^r p_t^{n|r} e^{-\ell_t^n}} + \frac{1}{N} p_{t+1}^{t+1} \\
&\overset{\text{Induction}}{=}\; \left(1 - p_{t+1}^{t+1}\right) \frac{u_t^n e^{-\ell_t^n}}{\sum_n u_t^n e^{-\ell_t^n}} + \frac{1}{N} p_{t+1}^{t+1} \;=\; u_{t+1}^n,
\end{aligned}$$

and find the Fixed Share update equation (5) for switching rate $\alpha_t = \frac{N-1}{N} p_t^t$. ■

Discussion. This unification points out that the Fixed Share algorithm can be viewed/implemented in three different ways. Depending on the situation, one of these may be more attractive. For example, Hazan and Seshadhri (2009) show that the FLH viewpoint scales up naturally to continuous optimization, and Luo and Schapire (2015) use our specialists viewpoint to design parameterless adaptive regret algorithms for the Hedge setting.

4. The Adaptive Regret of Fixed Share

We have seen in the previous section that both intuitive approaches to obtain algorithms with low adaptive regret result in Fixed Share. We take this convergence to mean that Fixed Share is the most fundamental adaptive algorithm. The tracking regret for Fixed Share is already well-studied. In Section 4.1 we thoroughly analyse the adaptive regret of Fixed Share. We obtain the worst-case adaptive regret for mix loss. This result implies the known tracking regret bounds. Then in Section 4.2 we characterise the achievable (by means of any algorithm) bounds on worst-case adaptive regret. We prove an information-theoretic lower bound for mix loss that must hold for any algorithm, and which is tight for Fixed Share. We show that the Pareto optimal bounds are exactly the Fixed Share bounds. This establishes Fixed Share as *the* answer for adaptive regret. Finally, in Section 4.3, we investigate the possibility of improving the adaptive regret for all "late" intervals by completely forgoing the regret guarantees on "early" intervals. We conclude that this is basically impossible.

4.1 The Exact Worst-case Mix-loss Adaptive Regret for Fixed Share

Define the *worst-case adaptive regret* on $[t_1, t_2]$ to be the supremum of (1) over all data sequences (cf. Definition 7 below). In this section we first compute the exact worst-case adaptive regret of Fixed Share with arbitrary switching rate $\alpha_t \in [0, \frac{N-1}{N}]$. Then we obtain certain regret bounds of interest, including the tracking regret bound, for particular choices of α_t.

Theorem 4 *The worst-case adaptive regret of Fixed Share with $\alpha_t \in [0, \frac{N-1}{N}]$ and with N experts on interval $[t_1, t_2]$ equals*

$$-\ln\left(\frac{\alpha_{t_1}}{N-1}\prod_{t=t_1+1}^{t_2}(1-\alpha_t)\right) \tag{9}$$

with the convention (7).

Proof The proof consists of two parts. First we claim that the worst-case data for the interval $[t_1, t_2]$ in the setting of Protocol 1 is rather simple: on the interval there is one *good* expert (all others get infinite losses) and on the single trial before the interval (if $t_1 > 1$) this expert suffers infinite loss while others do not. The proof of this fact can be found in Appendix B.

Now we will compute the regret on this data. The regret of Fixed Share on the interval $[t_1, t_2]$ is $-\ln$ of the product of the weights put on the good expert (say, n) on this interval:

$$R^{\mathrm{FS}}_{[t_1,t_2]} = -\ln\prod_{t_1\le t\le t_2} u_t^n\,.$$

It is straightforward to derive u_t^n from (5):

$$u_{t_1}^n \;=\; \frac{\alpha_{t_1}}{N-1} \qquad \text{and} \qquad u_t^n \;=\; 1-\alpha_t \qquad \text{for } t\in[t_1+1, t_2]$$

(this is also true when $t_1 = 1$); this implies the statement. ∎

Next we discuss the bounds resulting from three settings of the switching rate α_t.

4.1.1 Example 1: Constant Switching Rate

This is the original Fixed Share by Herbster and Warmuth (1998).

Corollary 5 *Fixed Share with constant switching rate $\alpha_t = \alpha$ for $t > 1$ (recall that $\alpha_1 = \frac{N-1}{N}$) has worst-case adaptive regret equal to*

$$\begin{aligned}
&\ln(N-1) - \ln\alpha - (t_2-t_1)\ln\bigl(1-\alpha\bigr) && \text{for } t_1 > 1, \text{ and}\\
&\ln N - (t_2-1)\ln\bigl(1-\alpha\bigr) && \text{for } t_1 = 1.
\end{aligned}$$

A slightly weaker upper bound was obtained by Cesa-Bianchi et al. (2012). The clear advantage of our analysis with equality is that we can obtain the standard Fixed Share

tracking regret bound by summing the above adaptive regret bounds on individual intervals. Comparing Fixed Share with the best sequence S of experts on the interval $[1,T]$ with m blocks, we obtain the bound

$$L^{\mathrm{FS}}_{[1,T]} - L^{S}_{[1,T]} \;\leq\; \ln N + (m-1)\ln(N-1) - (m-1)\ln\alpha - (T-m)\ln(1-\alpha)\,,$$

which is exactly the standard Fixed Share tracking bound (6). So we see that the reason why Fixed Share can effectively compete with switching sequences is that it can, in fact, effectively compete with any expert on any interval, that is, has small adaptive regret.

4.1.2 Example 2: Slowly Decreasing Switching Rate

The idea of slowly decreasing the switching rate was considered by Shamir and Merhav (1999) in the context of source coding, and later analysed for expert switching by Koolen and De Rooij (2008); we saw in Section 3.2 that it also underlies Follow the Leading History of Hazan and Seshadhri (2009). It results in tracking regret bounds that are almost as good as the bounds for constant α *with optimally tuned* α. These tracking bounds are again implied by the following corresponding adaptive regret bound.

Corollary 6 *Fixed Share with switching rate $\alpha_t = 1/t$ (except for $\alpha_1 = \frac{N-1}{N}$) has worst-case adaptive regret*

$$-\ln\left(\frac{1}{(N-1)t_1}\prod_{t=t_1+1}^{t_2}\frac{t-1}{t}\right) \;=\; \ln(N-1)+\ln t_2 \qquad \text{for } t_1>1, \text{ and} \tag{10a}$$

$$-\ln\left(\frac{1}{N}\prod_{t=2}^{t_2}\frac{t-1}{t}\right) \;=\; \ln N+\ln t_2 \qquad \text{for } t_1=1. \tag{10b}$$

Comparing Fixed Share with the best sequence S of experts on $[1,T]$ with m blocks we obtain, by summing the bound in Corollary 6 over all blocks,

$$L^{\mathrm{FS}}_{[1,T]} - L^{S}_{[1,T]} \;\leq\; \ln N + (m-1)\ln(N-1) + m\ln T. \tag{11}$$

For comparison, the bound for Fixed Share aggregated over the αs with a suitable prior is

$$L^{\mathrm{AFS}}_{[1,T]} - L^{S}_{[1,T]} \;\leq\; \ln N + (m-1)\ln(N-1) + m\ln T - \ln\big((m-1)!\big) \tag{12}$$

(see Vovk (1999), Theorem 2, setting $\eta := 1$, $k := m-1$, and $\epsilon := 1$). Our bound (11) is not as good as (12) in that the latter has the nonpositive (negative for $m>2$) addend $-\ln\big((m-1)!\big) \sim -m\ln m$. However, the difference does not appear great unless the number m of blocks is very large.[6] And whereas Fixed Share can be implemented in time $O(N)$ per trial, this aggregate seems to require work per-trial scaling with $\sqrt{t}$ (Monteleoni and Jaakkola, 2003; De Rooij and Van Erven, 2009).

6. If $m \leq T^c$ for some $c \in (0,1)$, then $(1-c)m\ln T \;\leq\; m\ln\frac{T}{m} \;\leq\; m\ln T$, so the block timing overheads of (11) and (12) differ by at most a constant factor.

4.1.3 Example 3: Quickly Decreasing Switching Rate

The bounds we have obtained so far depend on t_2 either linearly or logarithmically. To get bounds that depend on t_2 sub-logarithmically, or even not at all, one may instead decrease the switching rate faster than $1/t$, as analysed by Shamir and Merhav (1999) and Koolen and De Rooij (2013). To obtain a controlled trade-off, we consider setting the switching rate to $\alpha_t = \frac{1}{t \ln t}$, except for $\alpha_1 = \frac{N-1}{N}$ and, in the case $N \in \{2,3\}$, $\alpha_2 = \frac{N-1}{N}$ (this is needed since $\frac{1}{2\ln 2} \approx 0.72 > \frac{N-1}{N}$). This leads to adaptive regret at most

$$\ln(N-1) + \ln t_1 + \ln\ln t_1 - \sum_{t=t_1+1}^{t_2} \ln\left(1 - \frac{1}{t \ln t}\right) \;\leq\; \ln(N-1) + \ln t_1 + \ln\ln t_2 \tag{13a}$$

when $t_1 > 2$ or both $t_1 = 2$ and $N > 3$, at most

$$\ln N - \sum_{t=3}^{t_2} \ln\left(1 - \frac{1}{t \ln t}\right) \;\leq\; \ln N + \ln\ln t_2 + 0.37 \tag{13b}$$

when $t_1 = 2$ and $N \in \{2,3\}$, and at most

$$\ln N - \sum_{t=2}^{t_2} \ln\left(1 - \frac{1}{t \ln t}\right) \;\leq\; \ln N + \ln\ln t_2 + 1.65 \tag{13c}$$

when $t_1 = 1$ and $N > 3$ (remember that $\ln\ln 1$ is understood to be 0). In the case where $t_1 = 1$ and $N \in \{2,3\}$, the term $\frac{1}{2\ln 2}$ in (13c) (when it is present, i.e., when $t_2 > 1$) should be replaced by the smaller term $\frac{N-1}{N}$; this does not affect the validity of the bound. In all the cases, the bounds, (13a)–(13c), are stronger than (2b).

The dependence on t_2 in (13) is extremely mild. We can suppress it completely by increasing the dependence on t_1 just ever so slightly. If we set $\alpha_t = t^{-1-\epsilon}$, where $\epsilon > 0$, then the sum of the series $\sum_t \alpha_t$ is finite and the bound becomes

$$\ln(N-1) + (1+\epsilon)\ln t_1 + c_\epsilon \qquad \text{for } t_1 > 1, \text{ and} \tag{14a}$$

$$\ln N + c_\epsilon \qquad \text{for } t_1 = 1, \tag{14b}$$

where $c_\epsilon = -\sum_{t=2}^{\infty} \ln(1 - t^{-1-\epsilon})$. It is clear that the bound (14a) is far from optimal when t_1 is large: c_ϵ can be replaced by a quantity that tends to 0 as $O(t_1^{-\epsilon})$ as $t_1 \to \infty$. In particular, for $\epsilon = 1$ we have the bound

$$\ln N + 2\ln t_1 + \ln 2.$$

An interesting feature of this switching rate is that for the full interval $[t_1, t_2] = [1, T]$ the bound differs from the standard AA bound only by an additive term less than 1. In words, the overhead for small adaptive regret is negligible.

4.2 Fixed Share is Pareto Optimal for Adaptive Regret

We started by considering several intuitive constructions for adaptive algorithms, and saw that they all result in Fixed Share. We then obtained the worst-case adaptive regret of

Fixed Share. Intuitive as it may be, we have not answered the question whether Fixed Share is a good algorithm in the sense that its worst-case adaptive regret bounds are small. It is conceivable that there are smarter algorithms with better adaptive regret guarantees. See for example the palette of tracking algorithms proposed by Koolen and De Rooij (2013). And even if no better algorithms exist, there may still be algorithms that exhibit different trade-offs, in the sense that their worst-case adaptive regret is incomparable to that of Fixed Share.

So in this section we start from the other end and derive lower bounds that hold for any algorithm. As expected, we conclude that the bounds of Fixed Share (with any switching rate sequence $\alpha_t \leq \frac{N-1}{N}$) are Pareto optimal. But it came to us as a surprise that actually *all other bounds are strictly dominated*. No matter how smart the algorithm, its worst-case adaptive regret will be dominated by that of an instance of Fixed Share.

We call a mapping ϕ of intervals to regrets a *candidate guarantee*. Such a candidate guarantee is *realisable* if there is an algorithm for mix-loss prediction (Protocol 1) with adaptive regret at most ϕ. That is, we demand

$$R_{[t_1,t_2]} \;\leq\; \phi(t_1,t_2)$$

for all sequences of expert losses $\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots$ in $(-\infty,\infty]^N$ and all choices of the interval $1 \leq t_1 \leq t_2$. We say that a realizable guarantee ϕ *dominates* a realizable guarantee ψ if $\phi(t_1,t_2) \leq \psi(t_1,t_2)$ for all intervals $[t_1,t_2]$; and we say that ϕ *strictly dominates* ψ if, furthermore, $\phi(t_1,t_2) < \psi(t_1,t_2)$ for some interval $[t_1,t_2]$. A realisable guarantee ϕ is *Pareto optimal* if it is not strictly dominated by any realizable guarantee.

We are interested in Pareto-optimal guarantees. Every such guarantee is, by definition, the worst-case regret of an algorithm. Let us make that precise:

Definition 7 *We say that ϕ is the* worst-case adaptive regret *of a given algorithm if*

$$\phi(t_1,t_2) \;=\; \sup_{\boldsymbol{\ell}_1,\boldsymbol{\ell}_2,\ldots} R_{[t_1,t_2]} \qquad \text{for all } 1 \leq t_1 \leq t_2.$$

The main step in characterising the Pareto optimal guarantees is showing that worst-case adaptive regrets cannot be too small.

Theorem 8 *Let ϕ be the worst-case adaptive regret of some algorithm. Then*

$$\phi(t,t) \;\geq\; \ln N \qquad \text{for all } 1 \leq t \text{ and} \tag{15a}$$

$$\phi(t_1,t_2) \;\geq\; \phi(t_1,t_1) + \sum_{t=t_1+1}^{t_2} -\ln\left(1-(N-1)e^{-\phi(t,t)}\right) \qquad \text{for all } 1 \leq t_1 < t_2. \tag{15b}$$

Proof We first show (15a). Since at any time t the N weights played must sum to one, the smallest weight must be $\leq 1/N$. By hitting all others with infinite loss we force regret at least $\ln N$ over the interval $[t,t]$. Now suppose (15a) holds throughout, and consider any interval $[t_1,t_2]$. Fix $\epsilon > 0$. Let $\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t_1}$ be data on which the regret over $[t_1,t_1]$ exceeds $\phi(t_1,t_1) - \epsilon$. Let n^* be the (any) best expert in trial t_1. Now for all $t \in (t_1,t_2]$ choose $\ell_t^{n^*} = 0$ and $\ell_t^n = \infty$ for all $n \neq n^*$. In any trial t, the algorithm must allocate at least

weight $e^{-\phi(t,t)}$ to each expert to guarantee ϕ on the singleton interval $\{t\}$. Since the weights sum to one, the weight allocated to expert n^* during each trial $t \in (t_1, t_2]$ can be at most $1-(N-1)e^{-\phi(t,t)}$. Hence the loss of the algorithm must be at least $-\ln\bigl(1-(N-1)e^{-\phi(t,t)}\bigr)$. Since the loss of the best expert n^* on $(t_1, t_2]$ is zero, the regret is at least the right-hand side of (15b) minus ϵ. The worst-case regret $\phi(t_1, t_2)$ must be at least as large. Since this holds for all ϵ, we have proved (15b). ■

A realisable guarantee is witnessed by some algorithm, and is therefore dominated by the worst-case adaptive regret of that algorithm. We proved that this worst-case adaptive regret must satisfy (15). We now show that any guarantee satisfying (15) is realised by an instance of Fixed Share.

Theorem 9 *Let ϕ satisfy* (15). *Then Fixed Share with switching rate sequence $\alpha_2, \alpha_3, \ldots$ where $\alpha_t = (N-1)e^{-\phi(t,t)}$ guarantees ϕ.*

Proof From (15a) we know that $\alpha_t \leq (N-1)/N$, so the worst case regret of Fixed Share is given by Theorem 4. In particular (9) with our choice of α_t equals the right-hand side of (15b), and so FS guarantees ϕ. Note that the fact that α_1 is always set to the specific value $(N-1)/N$ only works in our favour here. ■

By combining the preceding two theorems, in the following two corollaries we characterize realizable guarantees and obtain canonical representatives of the Pareto guarantees for adaptive regret.

Corollary 10 *Let ϕ be a candidate guarantee. The following are equivalent:*

- *ϕ is realisable*
- *there exists $\psi \leq \phi$ such that ψ satisfies* (15)
- *ϕ is dominated by the worst-case adaptive regret of a Fixed Share.*

Before stating the second corollary we need to tackle the problem of big α_ts.

Lemma 11 *Fixed Share with $\alpha_t \in [0, \frac{N-1}{N}]$ is Pareto optimal. If $\sup_t \alpha_t > \frac{N-1}{N}$, Fixed Share with such parameters α_t is strictly dominated by Fixed Share with parameters $\alpha_t \wedge \frac{N-1}{N}$.*

Proof Suppose Fixed Share with parameters $\alpha_t \in [0, \frac{N-1}{N}]$ dominates Fixed Share with parameters $\beta_t \in [0, \frac{N-1}{N}]$. Equation (9) with $t_1 = t_2$ then implies $\alpha_t \geq \beta_t$ for all t. Combining this with (9) with $t_1 = 1$ now implies $\alpha_t = \beta_t$ for all t. In combination with the last statement of Corollary 10 this proves the first statement of the lemma.

For the second statement of the lemma, we should prove that in the case $\sup_t \alpha_t > \frac{N-1}{N}$ the worst-case regret of Fixed Share with parameters α_t for a fixed interval $[t_1, t_2]$ containing t with $\alpha_t > \frac{N-1}{N}$ will be greater than the worst-case regret of Fixed Share with parameters $\alpha_t \wedge \frac{N-1}{N}$) (given by (9) with α_t replaced by $\alpha_t \wedge \frac{N-1}{N}$). To see this, modify the "worst-case data" described in the proof of Theorem 4 (which are no longer worst-case for α_t) as follows: if $\alpha_{t_1} > \frac{N-1}{N}$ (and so $t_1 > 1$), make the loss of the good expert at step $t_1 - 1$ finite and

make the losses of all other experts at step $t_1 - 1$ infinite; if $\alpha_t > \frac{N-1}{N}$ for $t \in (t_1, t_2]$, make the loss of the good expert at step $t-1$ infinite and make the loss of another expert at step $t-1$ finite. ■

We conclude that Fixed Share is *the* answer for worst-case adaptive regret bounds.

Corollary 12 *Let ϕ be a candidate guarantee. The following are equivalent:*

- *ϕ is Pareto optimal*
- *ϕ satisfies* (15a) *with equality for $t = 1$ and satisfies* (15b) *with equality throughout*
- *ϕ is the worst-case adaptive regret of a Fixed Share with parameters $\alpha_t \in [0, \frac{N-1}{N}]$.*

Proof First, let ϕ be Pareto optimal. Since it is then the worst-case adaptive regret of some algorithm, it satisfies (15). By Theorem 9 ϕ is guaranteed by a Fixed Share with $\alpha_t \in [0, \frac{N-1}{N}]$, and so ϕ is the worst-case adaptive regret of such a Fixed Share. And, as noted in the proof of Theorem 9, the worst case adaptive regret of such a Fixed Share satisfies (15a) with equality for $t = 1$ and (15b) with equality throughout. Second, let ϕ satisfy (15a) with equality for $t = 1$ and (15b) with equality throughout. By Theorem 9 ϕ is guaranteed by Fixed Share with $\alpha_t = (N-1)e^{\phi(t,t)} \in [0, \frac{N-1}{N}]$. Now Theorem 4 implies that ϕ is the worst-case adaptive regret of this Fixed Share. Third, let ϕ be the worst-case adaptive regret of a Fixed Share with parameters $\alpha_t \in [0, \frac{N-1}{N}]$. Then ϕ is Pareto optimal by Lemma 11. ■

4.3 Sacrifice Early to Benefit Later? Impossible

We have seen that the Pareto optimal guarantees are those witnessed by Fixed Share. We saw several bounds (2), worked out in detail in Section 4.1, with different dependencies on the interval endpoints t_1 and t_2. In this section we investigate the possibility of forgoing completely the guarantees on early intervals to substantially improve the guarantees on all late intervals. We conclude that this is essentially impossible.

The main tool in this section is a slight relaxation of Theorem 8 that holds for all guarantees, not only for worst-case regrets.

Corollary 13 *If $\phi(t_1, t_2)$ is realisable, then*

$$\phi(t_1, t_2) \;\geq\; \ln N + \sum_{t=t_1+1}^{t_2} -\ln\left(1 - (N-1)e^{-\phi(t,t)}\right) \qquad \text{for all } 1 \leq t_1 \leq t_2. \tag{16}$$

Proof Plug (15a) into (15b) and notice that increasing ϕ increases the left-hand side and decreases the right-hand side of (16). ■

The following corollary shows that the stronger form (10) of (2a) is essentially tight: for large t_1 and t_2 we cannot even improve the right-hand side of (10a) by a positive constant (it is not sufficient to ignore all intervals with $t_1 < T_0$ for an arbitrarily large T_0).

Corollary 14 *Fix a constant $C < \ln(N-1)$ and an arbitrarily large positive integer constant T_0. The following guarantee is not realisable:*

$$\phi(t_1, t_2) \;=\; \begin{cases} C + \ln t_2 & \text{if } t_1 \geq T_0 \\ \infty & \text{otherwise.} \end{cases} \tag{17}$$

Proof For $t_1 \geq T_0 - 1$, the right hand side of (16) is at least (using $-\ln(1-x) \geq x$ and Euler's summation formula)

$$\ln N - \sum_{t=t_1+1}^{t_2} \ln\left(1 - \frac{N-1}{e^C t}\right) \;\geq\; \ln N + \frac{N-1}{e^C} \sum_{t=t_1+1}^{t_2} \frac{1}{t} \;\geq\; \ln N + \frac{N-1}{e^C}\left(\ln t_2 - \ln t_1 - D\right),$$

for some constant D. (The inequality $-\ln(1-x) \geq x$ assumes only $x < 1$ and so is applicable if T_0 is sufficiently large.) For a fixed t_1 (say, $t_1 = T_0$), this will exceed $C + \ln t_2$ (the guarantee 17) when t_2 is sufficiently large. Contradiction. ■

Our next corollary is about the tightness of (2b) and its elaboration (13) (especially 13a).

Corollary 15 *Fix a constant $C < \ln(N-1)$ and positive integer T_0. The following candidate guarantee is not realisable:*

$$\phi(t_1, t_2) \;=\; \begin{cases} C + \ln t_1 + \ln\ln t_2 & \text{if } t_1 \geq T_0 \\ \infty & \text{otherwise.} \end{cases}$$

Proof As in the previous proof, we can see that for $t_1 \geq T_0 - 1$ the right hand side of (16) is at least

$$\ln N - \sum_{t=t_1+1}^{t_2} \ln\left(1 - \frac{N-1}{e^C t \ln t}\right) \;\geq\; \ln N + \sum_{t=t_1+1}^{t_2} \frac{N-1}{e^C t \ln t} \;\geq\; \ln N + \frac{N-1}{e^C}\big(\ln\ln t_2 - \ln\ln t_1 - D\big).$$

For a fixed t_1 and a large enough t_2 this will exceed $C + \ln t_1 + \ln\ln t_2$. Contradiction. ■

Finally, we explore the tightness of (2c) and its elaboration given later in the paper: see (14) and the discussion afterwards. Let us say that a sequence $a(1), a(2), \ldots$ is $O(1)$-*realisable* if there is a C such that the candidate guarantee $\phi(t_1, t_2) = a(t_1) + C$ is realisable. Let us say that a sequence $a(1), a(2), \ldots$ is $O(1)$-*realisable in the long run* if there are C and T_0 such that the candidate guarantee

$$\phi(t_1, t_2) \;=\; \begin{cases} a(t_1) + C & \text{if } t_1 \geq T_0 \\ \infty & \text{otherwise} \end{cases} \tag{18}$$

is realisable.

Corollary 16 *A sequence $a(t)$ is $O(1)$-realisable*

- *if and only if it is $O(1)$-realisable in the long run, and*

- *if and only if* $\sum_t e^{-a(t)} < \infty$.

Proof Suppose $\sum_t e^{-a(t)} < \infty$. Then $a(t)$ is $O(1)$-realisable by Theorem 4 (set $\alpha_t = e^{-a(t)}$ from some t on). To prove that the series converges when $a(t)$ is $O(1)$-realisable in the long run, suppose $\sum_t e^{-a(t)} = \infty$. If (18) is realisable, we can again see that for $t_1 \geq T_0 - 1$ the right hand side of (16) is at least

$$\ln N + \sum_{t=t_1+1}^{t_2} -\ln\left(1 - (N-1)e^{-C}e^{-a(t)}\right) \;\geq\; \ln N + \sum_{t=t_1+1}^{t_2} (N-1)e^{-C}e^{-a(t)}.$$

For a fixed t_1 and a large enough t_2 this will exceed $a(t_1) + C$. Contradiction. ■

The first statement of Corollary 16 can be interpreted as a weak statement of impossibility to sacrifice early in order to benefit later: we can gain at most a constant when we sacrifice early. (And we saw below (14) that we can indeed gain a constant.) This statement, however, is very general, as the following simple argument shows.

Let us say that a candidate guarantee $\phi(t_1, t_2)$ is $O(1)$-*realisable* if there is a C such that the candidate guarantee $\psi(t_1, t_2) = \phi(t_1, t_2) + C$ is realisable. Let us say that the candidate guarantee $\phi(t_1, t_2)$ is $O(1)$-*realisable in the long run* if there are C and T_0 such that the candidate guarantee

$$\psi(t_1, t_2) \;=\; \begin{cases} \phi(t_1, t_2) + C & \text{if } t_1 \geq T_0 \\ \infty & \text{otherwise} \end{cases} \tag{19}$$

is realisable.

Lemma 17 *A candidate guarantee ϕ is $O(1)$-realisable if and only if it is $O(1)$-realisable in the long run.*

Proof Suppose ϕ is $O(1)$-realisable in the long run and let C and T_0 be such that (19) is realisable. Aggregate using the AA the following prediction strategies: a prediction algorithm that realises ϕ; a prediction algorithm suffering a bounded regret over all $[1, t]$; a prediction algorithm suffering a bounded regret over all $[2, t]$;...; a prediction algorithm suffering a bounded regret over all $[T_0 - 1, t]$. ■

5. Conclusion

We examined the problem of guaranteeing small adaptive regret for the setting of prediction with expert advice. In the first part we considered two techniques to obtain adaptive algorithms: using virtual specialist experts and restarting classical algorithms. We showed that both result in Fixed Share with a variable switching rate. In the second part we computed the exact worst-case adaptive regret for Fixed Share, thus tightening the existing upper bounds. So much, in fact, that by summing these worst-case regrets over a partition of the interval $[1, T]$ we recover the standard Fixed Share tracking bound. In other words, the tracking performance of Fixed Share is a consequence of its adaptivity.

Protocol 2 Prediction with Expert Advice

for $t = 1, 2, \ldots$ **do**
  Expert n announces prediction $\gamma_t^n \in \Gamma$ for each $n = 1, \ldots, N$
  Learner announces prediction $\gamma_t \in \Gamma$
  Reality announces outcome $\omega_t \in \Omega$
  Learner suffers loss $\lambda(\gamma_t, \omega_t)$
end for

We then give an information-theoretic characterisation of the achievable worst-case adaptive regrets, and conclude that the Pareto optimal regrets are exactly the Fixed Share regrets. Fixed Share simply is the optimal adaptive algorithm.

Open problem. Whereas upper bounds readily transfer to mixable losses, obtaining adaptive regret lower bounds for mixable losses is much more tricky. It is fair to call the lower bound argument by Vovk (1998) for classical regret complicated, and this would be a special case for adaptive regret lower bounds.

Acknowledgments

First author supported by Veterinary Laboratories Agency (VLA) of Department for Environment, Food and Rural Affairs (Defra) and by a Leverhulme Trust research project grant RPG-2013-047 "On-line Self-Tuning Learning Algorithms for Handling Historical Information". Second author supported by NWO Rubicon grant 680-50-1010. Third author supported by EPSRC grant EP/I030328/1 and AFOSR grant "Semantic Completions".

Appendix A. Adaptive Regret for Mix Loss Transfers to Mixable Losses

The protocol of prediction with expert advice is given as Protocol 2. Predictions are made sequentially, their quality is measured by the loss function λ and Learner has access to a (finite) pool of N experts. An important class of loss functions that allow for effective algorithms are mixable losses.

Definition 18 *A loss function λ is called* η-mixable*, where $\eta > 0$, if for every N, every sequence of predictions $\gamma^1, \ldots, \gamma^N$ and every sequence of normalised nonnegative weights $u^1, \ldots, u^N$ there exists a prediction $\gamma \in \Gamma$ such that for every outcome $\omega \in \Omega$*

$$\lambda(\gamma, \omega) \;\leq\; -\frac{1}{\eta} \ln \left(\sum_{n=1}^{N} u^n e^{-\eta \lambda(\gamma^n, \omega)} \right). \tag{20}$$

A function Σ that maps every sequence of predictions $\gamma^1, \ldots, \gamma^N$ and every sequence of normalised nonnegative weights $u^1, \ldots, u^N$ of the same length to $\gamma \in \Gamma$ satisfying (20) is called an η-perfect substitution function*.*

To establish mixability it is sufficient to verify (20) for $N = 2$. Examples of mixable games are discussed by Vovk (2001). Note that $1/\eta$-scaled mix loss is the baseline used in

the definition of mixability: see (20). In this sense the mix loss is the hardest mixable loss. It is hence no surprise that adaptive regret bounds for mix loss immediately transfer to any mixable loss:

Fact 19 *Let X be a mix-loss algorithm with worst-case adaptive regret $\phi(t_1,t_2)$. If λ is η-mixable then there is an algorithm Y with adaptive regret at most $\phi(t_1,t_2)/\eta$.*

Proof Let Σ be an η-perfect substitution function for λ. We choose Y to be the algorithm that operates as follows. At each trial $t=1,2,\dots$ it obtains prediction $\boldsymbol{u}_t$ from X, predicts with $\gamma_t = \Sigma(\boldsymbol{u}_t, \boldsymbol{\gamma}_t)$, and feeds into X losses $\ell_t^n = \eta\lambda(\gamma_t^n, \omega_t)$ (from the point of view of Y these are scaled losses rather than losses). Then for each interval $[t_1,t_2]$ and reference expert n we have

$$\begin{aligned} L^Y_{[t_1,t_2]} - L^n_{[t_1,t_2]} &= \sum_{t=t_1}^{t_2} \lambda(\gamma_t,\omega_t) - \sum_{t=t_1}^{t_2} \lambda(\gamma_t^n,\omega_t) \\ &\le -\frac{1}{\eta}\sum_{t=t_1}^{t_2} \ln \sum_j u_t^j e^{-\eta\lambda(\gamma_t^j,\omega_t)} - \sum_{t=t_1}^{t_2} \lambda(\gamma_t^n,\omega_t) \\ = -\frac{1}{\eta}\sum_{t=t_1}^{t_2} \ln \sum_j u_t^j e^{-\ell_t^j} - \frac{1}{\eta}\sum_{t=t_1}^{t_2} \ell_t^n &= \frac{1}{\eta}\left(L^X_{[t_1,t_2]} - L^n_{[t_1,t_2]}\right) \le \frac{1}{\eta}\phi(t_1,t_2), \end{aligned}$$

where the first inequality follows from the definition of an η-perfect substitution function and the last one from our assumption about X. ∎

Fact 19 shows that all our performance guarantees for the mix-loss protocol carry over to the protocol of prediction with expert advice with a mixable loss function.

Appendix B. Worst-case Adaptive Regret Data for Fixed Share

In this subsection we prove that the worst-case data for Fixed Share has the following form. On the interval $[t_1,t_2]$ we are interested in all but one expert suffer infinite loss and on the step preceding t_1 (if $t_1 \neq 1$) this one expert suffers infinite loss himself. The construction is iterative and we start constructing the data from the end of the interval.

Lemma 20 *For any history prior to the step t_2 the adaptive regret $R^n_{[t_1,t_2]}$ w.r.t. expert n on the interval $[t_1,t_2]$ is maximised with $\ell^k_{t_2} = \infty$ for $k \neq n$.*

Proof Let us differentiate the adaptive regret w.r.t. $\ell^k_{t_2}$:

$$\frac{\partial R^n_{[t_1,t_2]}}{\partial \ell^k_{t_2}} = \frac{\partial(\ell_{t_2} - \ell^n_{t_2})}{\partial \ell^k_{t_2}} = -\frac{\partial}{\partial \ell^k_{t_2}} \ln \sum_j u^j_{t_2} e^{-\ell^j_{t_2}} - \mathbf{1}_{\{n=k\}} = \frac{u^k_{t_2} e^{-\ell^k_{t_2}}}{\sum_j u^j_{t_2} e^{-\ell^j_{t_2}}} - \mathbf{1}_{\{n=k\}}.$$

This is positive for all $k \neq n$ and becomes zero for $k=n$ when we plug in $\ell^k_{t_2} = \infty$ for those. ∎

Lemma 21 *Consider switching rates $\alpha_t \in [0, \frac{N-1}{N}]$. Fix a comparator expert n. Let $t \in [t_1, t_2]$. Suppose that the losses for steps $s = t+1, \ldots, t_2$ satisfy $\ell_s^k = \infty$ for $k \neq n$. Then for the adaptive regret $R^n_{[t_1,t_2]}$ is maximised with $\ell_t^k = \infty$ for $k \neq n$.*

Proof Let us start with showing that on the steps $t+1$ and $t+2$ the data is organised as we want to, that is, the n-th expert is good and all others suffer infinite loss, then Learner's loss on step $t+2$ is not dependent on what happens at time t and before. This follows immediately from (5), as

$$\ell_{t+2} \;=\; -\ln\left(1-\alpha_{t+2}\right) .$$

Now let us differentiate the adaptive regret $R^n_{[t_1,t_2]}$ w.r.t. ℓ_t^k assuming that the future losses are set up as we want. Let us show that the derivatives w.r.t. ℓ_t^k where $k \neq n$ are all positive. For those,

$$\frac{\partial R^n_{[t_1,t_2]}}{\partial \ell_t^k} \;=\; \frac{\partial \ell_t}{\partial \ell_t^k} + \frac{\partial \ell_{t+1}}{\partial \ell_t^k} .$$

Expanding the second one gives (as before, $k \neq n$):

$$\begin{aligned}
\frac{\partial \ell_{t+1}}{\partial \ell_t^k} \;&=\; \frac{\partial}{\partial \ell_t^k} - \ln\left(\frac{\alpha_{t+1}}{N-1} + (1 - \frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t - \ell_t^n}\right) \\
&=\; -\frac{(1-\frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t-\ell_t^n} \frac{\partial}{\partial \ell_t^k}\ell_t}{\frac{\alpha_{t+1}}{N-1} + (1-\frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t-\ell_t^n}} .
\end{aligned}$$

So we see that

$$\begin{aligned}
\frac{\partial R^n_{[t_1,t_2]}}{\partial \ell_t^k} \;&=\; \frac{\partial \ell_t}{\partial \ell_t^k}\left(1 - \frac{(1-\frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t-\ell_t^n}}{\frac{\alpha_{t+1}}{N-1} + (1-\frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t-\ell_t^n}}\right) \\
&=\; \frac{\partial \ell_t}{\partial \ell_t^k}\left(\frac{\frac{\alpha_{t+1}}{N-1}}{\frac{\alpha_{t+1}}{N-1} + (1-\frac{N}{N-1}\alpha_{t+1}) u_t^n e^{\ell_t-\ell_t^n}}\right) \;>\; 0 .
\end{aligned}$$

So our worst-case pattern of losses extends one trial backwards. ■

Finally, we need to state the almost obvious fact that in order to maximise the adaptive regret we need to insert an infinite loss for the comparator expert right before the start of the interval, thus killing all the previous weight on him.

Lemma 22 *Consider switching rates $\alpha_t \in [0, \frac{N-1}{N}]$. Fix a comparator expert n. Suppose that the losses for steps $s = t_1, \ldots, t_2$ satisfy $\ell_s^k = \infty$ for $k \neq n$. Then the adaptive regret $R^n_{[t_1,t_2]}$ is maximised with $\ell^n_{t-1} = \infty$.*

Proof As before, the adaptive regret on steps starting from t_1+1 does not depend on $\ell^k_{t_1-1}$. So let us show that $\frac{\partial R^n_{[t_1,t_2]}}{\partial \ell^n_{t_1-1}} > 0$. We can reuse the proofs of previous lemmas for that:

$$\frac{\partial R^n_{[t_1,t_2]}}{\partial \ell^n_{t_1-1}} \;=\; \frac{\partial \ell_{t_1}}{\partial \ell^n_{t_1-1}} \;=\; -\frac{(1-\frac{N}{N-1}\alpha_{t_1}) u^n_{t_1-1} e^{\ell_{t_1-1}-\ell^n_{t_1-1}}}{\frac{\alpha_{t_1}}{N-1} + (1-\frac{N}{N-1}\alpha_{t_1}) u^n_{t_1-1} e^{\ell_{t_1-1}-\ell^n_{t_1-1}}} \frac{\partial\left(\ell_{t_1-1} - \ell^n_{t_1-1}\right)}{\partial \ell^n_{t_1-1}} \;>\; 0 ,$$

since $\frac{\partial\left(\ell_{t_1-1}-\ell^n_{t_1-1}\right)}{\partial \ell^n_{t_1-1}}$ is negative as follows from the proof of Lemma 20. ■

References

Olivier Bousquet and Manfred K. Warmuth. Tracking a small set of experts by mixing past posteriors. *Journal of Machine Learning Research*, 3:363–396, 2002.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Nicolò Cesa-Bianchi, Pierre Gaillard, Gábor Lugosi, and Gilles Stoltz. Mirror Descent meets Fixed Share (and feels no regret). In *NIPS Proceedings*, pages 989–997, 2012.

Alexey Chernov and Vladimir Vovk. Prediction with expert evaluators' advice. In *ALT Proceedings*, volume 5809 of *Lecture Notes in Computer Science*, pages 8–22, 2009.

Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.

Yoav Freund, Robert E. Schapire, Yoram Singer, and Manfred K. Warmuth. Using and combining predictors that specialize. In *STOC Proceedings*, pages 334–343, 1997.

András György, Tamás Linder, and Gábor Lugosi. Efficient tracking of large classes of experts. *IEEE Transactions on Information Theory*, 58(11):6709–6725, 2012.

Elad Hazan and Comandur Seshadhri. Adaptive algorithms for online optimization. In *Electronic Colloquium on Computational Complexity*, 2007. TR07-88.

Elad Hazan and Comandur Seshadhri. Efficient learning algorithms for changing environments. In *ICML Proceedings*, pages 393–400, 2009.

Mark Herbster and Manfred K. Warmuth. Tracking the best expert. *Machine Learning*, 32:151–178, 1998.

Wouter M. Koolen and Steven de Rooij. Combining expert advice efficiently. In *COLT Proceedings*, pages 275–286, 2008.

Wouter M. Koolen and Steven de Rooij. Universal codes from switching strategies. *IEEE Transactions on Information Theory*, 59(11):7168–7185, November 2013.

Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108:212–261, 1994.

Haipeng Luo and Robert E. Schapire. Achieving all with no parameters: Adaptive NormalHedge. In *COLT Proceedings*, pages 1286–1304, June 2015.

Claire Monteleoni and Tommi Jaakkola. Online learning of non-stationary sequences. In *NIPS Proceedings*, pages 1093–1100, 2003.

Steven de Rooij and Tim van Erven. Learning the switching rate by discretising Bernoulli sources online. In *AISTATS Proceedings*, pages 432–439, 2009.

Gil I. Shamir and Neri Merhav. Low complexity sequential lossless coding for piecewise stationary memoryless sources. *IEEE Transactions on Information Theory*, 45:1498–1519, 1999.

Vladimir Vovk. Aggregating strategies. In *COLT Proceedings*, pages 371–383, 1990.

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

Vladimir Vovk. Derandomizing stochastic prediction strategies. *Machine Learning*, 35: 247–282, 1999.

Vladimir Vovk. Competitive on-line statistics. *International Statistical Review*, 69:213–248, 2001.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *ICML Proceedings*, pages 928–936, 2003.