Efficient tracking of a growing number of experts

Jaouad Mourtada, Odalric-Ambrym Maillard

Journal of Machine Learning Research 76:1–23, 2017 Algorithmic Learning Theory 2017

Efficient tracking of a growing number of experts

Jaouad Mourtada JAOUAD.MOURTADA@POLYTECHNIQUE.EDU
Centre de Mathématiques Appliquées
École Polytechnique
91128 Palaiseau, France

Odalric-Ambrym Maillard ODALRIC.MAILLARD@INRIA.FR
Inria Lille - Nord Europe
59650 Villeneuve d'Ascq, France

Editors: Steve Hanneke and Lev Reyzin

Abstract

We consider a variation on the problem of prediction with expert advice, where new forecasters that were unknown until then may appear at each round. As often in prediction with expert advice, designing an algorithm that achieves near-optimal regret guarantees is straightforward, using aggregation of experts. However, when the comparison class is sufficiently rich, for instance when the best expert and the set of experts itself changes over time, such strategies naively require to maintain a prohibitive number of weights (typically exponential with the time horizon). By contrast, designing strategies that both achieve a near-optimal regret and maintain a reasonable number of weights is highly non-trivial. We consider three increasingly challenging objectives (simple regret, shifting regret and sparse shifting regret) that extend existing notions defined for a fixed expert ensemble; in each case, we design strategies that achieve tight regret bounds, adaptive to the parameters of the comparison class, while being computationally inexpensive. Moreover, our algorithms are anytime, agnostic to the number of incoming experts and completely parameter-free. Such remarkable results are made possible thanks to two simple but highly effective recipes: first the "abstention trick" that comes from the *specialist* framework and enables to handle the least challenging notions of regret, but is limited when addressing more sophisticated objectives. Second, the "muting trick" that we introduce to give more flexibility. We show how to combine these two tricks in order to handle the most challenging class of comparison strategies.

Keywords: Online learning; Prediction with expert advice; Shifting regret; Anytime strategies.

1. Introduction

Aggregation of experts is a well-established framework in machine learning (Cesa-Bianchi and Lugosi, 2006; Vovk, 1998; Györfi et al., 1999; Haussler et al., 1998), that provides a sound strategy to combine the forecasts of many different sources. This is classically considered in the sequential prediction setting, where at each time step, a learner receives the predictions of experts, uses them to provide his own forecast, and then observes the true value of the signal, which determines his loss and those of the experts. The goal is then to minimize the *regret* of the learner, which is defined as the difference between his cumulated loss and that of the best expert (or combination thereof), no matter what the experts' predictions or the values of the signal are.

A standard assumption in the existing literature is that the set of experts is known before the beginning of the game. In many situations, however, it is desirable to add more and more forecasters over time. For instance, in a non-stationary setting one could add new experts trained on a

fraction of the signal, possibly combined with change point detection. Even in a stationary setting, a growing number of increasingly complex models enables to account for increasingly subtle properties of the signal without having to include them from the start, which can be needlessly costly computationally (as complex models, which take more time to fit, are not helpful in the first rounds) or even intractable in the case of an infinite number of models with no closed form expression. Additionally, in many realistic situations some completely novel experts may appear in an unpredicted way (possibly due to innovation, the discovery of better algorithms or the availability of new data), and one would want a way to safely incorporate them to the aggregation procedure.

In this paper, we study how to amend aggregation of experts strategies in order to incorporate novel experts that may be added on the fly at any time step. Importantly, since we do not know in advance when new experts are made available, we put a strong emphasis on *anytime* strategies, that do not assume the time horizon is finite and known. Likewise, our algorithms should be agnostic to the total number of experts available at a given time. Three notions of regret of increasing complexity will be defined for growing expert sets, that extend existing notions to a growing expert set. Besides comparing against the best expert, it is natural in a growing experts setting to track the best expert; furthermore, when the number of experts gets large, it becomes profitable to track the best expert in a small pool of good experts. For each notion, we propose corresponding algorithms with tight regret bounds. As is often the case in structured aggregation of experts, the key difficulty is typically not to derive the regret bounds, but to obtain efficient algorithms. All our methods exhibit minimal time and space requirements that are linear in the number of present experts.

Related work. This work builds on the setting of prediction with expert advice (Cesa-Bianchi and Lugosi, 2006; Vovk, 1998; Herbster and Warmuth, 1998) that originates from the work on universal prediction (Ryabko, 1984, 1988; Merhav and Feder, 1998; Györfi et al., 1999). We make use of the notion of *specialists* (Freund et al., 1997; Chernov and Vovk, 2009) and its application to *sleeping experts* (Koolen et al., 2012), as well as the corresponding standard extensions (Fixed Share, Mixing Past Posteriors) of basic strategies to the problem of *tracking the best expert* (Herbster and Warmuth, 1998; Koolen and de Rooij, 2013; Bousquet and Warmuth, 2002); see also Willems (1996); Shamir and Merhav (1999) for related work in the context of lossless compression. Note that, due to its versatility, aggregation of experts has been adapted successfully to a number of applications (Monteleoni et al., 2011; McQuade and Monteleoni, 2012; Stoltz, 2010). It should be noted that the literature on prediction with expert advice is split in two categories: the first one focuses on exp-concave loss functions, whereas the second studies convex bounded losses. While our work belongs to the first category, it should be possible to transport our regret bounds to the convex bounded case by using time-varying learning rates, as done e.g. by Hazan and Seshadhri (2009) and Gyorgy et al. (2012). In this case, the growing body of work on the automatic tuning of the learning rate (de Rooij et al., 2014; Koolen et al., 2014) as well as alternative aggregation schemes (Wintenberger, 2017; Koolen and van Erven, 2015; Luo and Schapire, 2015) might open the path for even further improvements.

The use of a growing expert ensemble was already proposed by Györfi et al. (1999) in the context of sequentially predicting an ergodic stationary time series, where new higher order Markov experts were introduced at exponentially increasing times (and the weights were reset as uniform); since consistency was the core focus of the paper, this simple "doubling trick" could be used, something we cannot afford when new experts arrive more regularly. Closer to our approach, growing expert ensembles have been considered in contexts where the underlying signal may be non-stationary, see e.g. Hazan and Seshadhri (2009); Shalizi et al. (2011). Of special interest to our problem is Shal-

izi et al. (2011), which considers the particular case when one new expert is introduced every τ time steps, and propose a variant of the Fixed Share (FS) algorithm analogous to our Growing-MarkovHedge algorithm. However, their algorithms depend on parameters which have to be tuned depending on the parameters of the comparison class, whereas our algorithms are parameter-free and do not assume the prior knowledge of the comparison class. Moreover, we introduce several other algorithms tailored to different notions of regret; in particular, we address the problem of comparing to sequences of experts that alternate between a small number of experts, a refinement that is crucial when the total set of experts grows, and has not been obtained previously in this context.

Another related setting is that of "branching experts" considered by Gofer et al. (2013), where each incumbent expert is split into several experts that may diverge later on. Their results include a regret bound in terms of the number of *leading experts* (whose cumulated loss was minimal at some point). Our approach differs in that it does not assume such a tree-like structure: a new entering forecaster is not assumed to be associated to an incumbent expert. More importantly, while Gofer et al. (2013) compare to the leaders in terms of cumulated loss (since the beginning of the game), our methods compete instead with sequences of experts that perform well on some periods, but can predict arbitrarily bad on others; this is harder, since the loss of the optimal sequence of experts can be significantly smaller than that of the best expert.

Outline. Our paper is organized as follows. After introducing the setting, notations and the different comparison classes, we provide in Section 2 an overview of our results, stated in less general but more directly interpretable forms. Then, Section 3 introduces the exponential weights algorithm and its regret, a classical preliminary result that will be used throughout the text. Sections 4, 5 and 6 form the core of this paper, and have the same structure: a generic result is first stated in the case of a fixed set of experts, before being turned into a strategy in the growing experts framework. Section 4 starts with the related *specialist* setting and adapts the algorithm into an anytime growing experts algorithm, with a more general formulation and regret bound involving *unnormalized priors*. Section 5 proposes an alternative approach, which casts the growing experts problem as one of competing against *sequences* of experts; this approach proves more flexible and general for our task, but perhaps surprisingly we can also recover algorithms that are essentially equivalent to the aggregation of growing experts with an unnormalized prior. Finally, the two approaches are combined in Section 6 in the context of *sleeping experts*, where we reinterpret the algorithm of Koolen et al. (2012) and extend it to more general priors before adapting it to the growing experts setting.

2. Overview of the results

Our work is framed in the classical setting of *prediction with expert advice* (Vovk, 1998; Cesa-Bianchi and Lugosi, 2006), which we adapt to account for a growing number of experts. The problem is characterized by its *loss function* $\ell : \mathscr{X} \times \mathscr{Y} \to \mathbf{R}$, where $\mathscr{X}$ is a convex *prediction space*, and $\mathscr{Y}$ is the *signal space*.

Let M_t be the total number of experts at time t, and $m_t = M_t - M_{t-1}$ be the number of experts introduced at time t. We index experts by their entry order, so that expert i is the i^{th} introduced expert and denote $\tau_i = \min\{t \geqslant 1 : i \leqslant M_t\}$ its *entry time* (the time at which it is introduced). We say we are in the *fixed expert set* case when $M_t = M$ for every $t \geqslant 1$ and in the *growing experts setting* otherwise. At each step $t \geqslant 1$, the experts $i = 1, \dots, M_t$ output their predictions $x_{i,t} \in \mathscr{X}$, which the learner uses to build $x_t \in \mathscr{X}$; then, the environment decides the value of the signal $y_t \in \mathscr{Y}$, which sets the losses $\ell_t = \ell(x_t, y_t)$ of the learner and $\ell_{i,t} = \ell(x_{i,t}, y_t)$ of the experts.

Notations. Let $\mathscr{P}_M$ be the *probability simplex*, *i.e.* the set of probability measures over the set of experts $\{1, \dots, M\}$. We denote by $\Delta(\cdot \,\|\, \cdot)$ the *Kullback-Leibler divergence*, defined for $\boldsymbol{u}, \boldsymbol{v} \in \mathscr{P}_M$ by $\Delta(\boldsymbol{u} \,\|\, \boldsymbol{v}) = \sum_{i=1}^M u_i \log \frac{u_i}{v_i} \geqslant 0$.

Loss function. Throughout this text, we make the following standard assumption[1] on the loss function (Cesa-Bianchi and Lugosi, 2006).

Assumption 1 *The loss function ℓ is* η-exp-concave *for some $\eta > 0$, in the sense that $\exp(-\eta\, \ell(\cdot, y))$ is concave on $\mathscr{X}$ for every observation $y \in \mathscr{Y}$. This is equivalent to the inequality*

$$\ell\left(\sum_{i=1}^M v_i\, x_i, y\right) \leqslant -\frac{1}{\eta} \log \sum_{i=1}^M v_i\, e^{-\eta\, \ell(x_i, y)} \tag{1}$$

for every $y \in \mathscr{Y}$, $\boldsymbol{x} = (x_i)_{1 \leqslant i \leqslant M} \in \mathscr{X}^M$ and $\boldsymbol{v} = (v_i)_{1 \leqslant i \leqslant M} \in \mathscr{P}_M$.

Remark 1 *An important example in the case when $\mathscr{X}$ is the set of probability measures over $\mathscr{Y}$ is the* logarithmic *or* self-information *loss $\ell(x, y) = -\log x(\{y\})$ for which the inequality holds with $\eta = 1$, and is actually an equality. Another example of special interest is the quadratic loss on a bounded interval: indeed, for $\mathscr{X} = \mathscr{Y} = [a, b] \subset \mathbf{R}$, $\ell(x, y) = (x - y)^2$ is $\frac{1}{2(b-a)^2}$-exp-concave.*

Several notions of regret can be considered in the growing expert setting. We review here three of them, each corresponding to a specific comparison class; we show the kind of bounds that our algorithms achieve, to illustrate the more general results stated in the subsequent sections. We provide more uniform bounds in Appendix E, and compare them with information-theoretic bounds.

Constant experts. Since the experts only output predictions after their entry time, it is natural to consider the *regret* with respect to each expert $i \geqslant 1$ over its time of activity, namely the quantity

$$\sum_{t=\tau_i}^T (\ell_t - \ell_{i,t}) \tag{2}$$

for every $T \geqslant \tau_i$. Note that this is equivalent to controlling (2) for every $T \geqslant 1$ and $i \leqslant M_T$. Algorithm GrowingHedge is particularly relevant in this context; with the choice of (unnormalized) prior weights $\pi_i = \frac{1}{\tau_i m_{\tau_i}}$, it achieves the following regret bound: for every $T \geqslant 1$ and $i \leqslant M_T$,

$$\sum_{t=\tau_i}^T (\ell_t - \ell_{i,t}) \leqslant \frac{1}{\eta} \log m_{\tau_i} + \frac{1}{\eta} \log \tau_i + \frac{1}{\eta} \log(1 + \log T)\,. \tag{3}$$

This bound has the merit of being simple, virtually independent of T and independent of the number of experts $(m_t)_{t > \tau_i}$ added after i. Several other instantiations of the general regret bound of GrowingHedge (Theorem 3) are given in Section 4.2.

Sequences of experts. Another way to study growing expert sets is to view them through the lens of sequences of experts. Given a sequence of experts $i^T = (i_1, \dots, i_T)$, we measure the performance of a learning algorithm against it in terms of the *cumulative regret*:

$$L_T - L_T(i^T) = \sum_{t=1}^T \ell_t - \sum_{t=1}^T \ell_{i_t, t}\,, \tag{4}$$

In order to derive meaningful regret bounds, some constraints have to be imposed on the comparison sequence; hence, we consider in the sequel different types of comparison classes that lead to different notions of regret, from the least to the most challenging one:

1. This could be readily replaced (up to some cosmetic changes in the statements and their proofs) by the more general η-*mixability* condition (Vovk, 1998), that allows to use higher learning rates η for some loss functions (such as the square loss, but not the logarithmic loss) by using more sophisticated combination functions.

(a) Sequences of fresh experts. These are *admissible* sequences of experts i^T, in the sense that $i_t \leqslant M_t$ for $1 \leqslant t \leqslant T$ (so that $\ell_{i_t,t}$ is always well-defined) that only switch to *fresh* (newly entered) experts, *i.e.* if $i_t \neq i_{t-1}$, then $M_{t-1}+1 \leqslant i_t \leqslant M_t$. More precisely, for each $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_k)$ with $1 < \sigma_1 < \cdots < \sigma_k \leqslant T$, $\mathscr{S}_T^{(f)}(\boldsymbol{\sigma})$ denotes the set of sequences of fresh experts whose only shifts occur at times $\sigma_1, \dots, \sigma_k$. Both the switch times $\boldsymbol{\sigma}$ and the number of shifts k are assumed to be unknown, although to obtain controlled regret one typically needs $k \ll T$. Comparing to sequences of fresh experts is essentially equivalent to comparing against constant experts; algorithms GrowingHedge and FreshMarkovHedge with $\pi_i = \frac{1}{m_{\tau_i}}$ achieve, for *every* $T \geqslant 1$, $k \leqslant T-1$ and $\boldsymbol{\sigma} = (\sigma_j)_{1 \leqslant j \leqslant k}$ (Theorems 3 and 6):

$$L_T - \inf_{i^T \in \mathscr{S}_T^{(f)}(\boldsymbol{\sigma})} L_T(i^T) \leqslant \frac{1}{\eta} \left\{ \log m_1 + \sum_{j=1}^{k} (\log m_{\sigma_j} + \log \sigma_j) + \log T \right\} \tag{5}$$

In particular, the regret with respect to any sequence of fresh experts with k shifts is bounded by $\frac{1}{\eta}\left((k+1)\log \max_{1 \leqslant t \leqslant T} m_t + (k+1)\log T\right)$.

(b) Arbitrary admissible sequences of experts. Like before, these are admissible sequences of experts that are piecewise constant with a typically small number of shifts k, except that shifts to *incumbent* (previously introduced) experts $i_t \leqslant M_{t-1}$ are now authorized. Specifically, given $\boldsymbol{\sigma}^0 = (\sigma_1^0, \dots, \sigma_{k_0}^0)$ and $\boldsymbol{\sigma}^1 = (\sigma_1^1, \dots, \sigma_{k_1}^1)$, we denote by $\mathscr{S}_T^{(a)}(\boldsymbol{\sigma}^0; \boldsymbol{\sigma}^1)$ the class of admissible sequences whose switches to fresh (resp. incumbent) experts occur only at times $\sigma_1^0 < \cdots < \sigma_{k_0}^0$ (resp. $\sigma_1^1 < \cdots < \sigma_{k_1}^1$). By Theorem 7, algorithm GrowingMarkovHedge with $\pi_i = \frac{1}{m_{\tau_i}}$ and $\alpha_t = \frac{1}{t}$ satisfies, for *every* $T \geqslant 1$, k_0, k_1 with $k_0 + k_1 \leqslant T-1$ and $\boldsymbol{\sigma}^0, \boldsymbol{\sigma}^1$:

$$L_T - \inf_{i^T \in \mathscr{S}_T^{(a)}(\boldsymbol{\sigma}^0; \boldsymbol{\sigma}^1)} L_T(i^T) \leqslant \frac{1}{\eta} \left\{ \log m_1 + \sum_{j=1}^{k} (\log m_{\sigma_j} + \log \sigma_j) + \sum_{j=1}^{k_1} \log \sigma_j^1 + 2\log T \right\} \tag{6}$$

where $k = k_0 + k_1$ and $\sigma_1 < \cdots < \sigma_k$ denote *all* shifts (either in $\boldsymbol{\sigma}^0$ or in $\boldsymbol{\sigma}^1$). Note that the upper bound (6) may be further relaxed as $\frac{1}{\eta}\left((k+1)\log \max_{1 \leqslant t \leqslant T} m_t + (k_0 + 2k_1 + 2)\log T\right)$.

(c) Sparse sequences of experts. These are admissible sequences i^T of experts that are additionally *sparse*, in the sense that they alternate between a small number $n \ll M_T$ of experts; again, n may be unknown in advance. Denoting $\mathscr{S}_T^{(s)}(\boldsymbol{\sigma}, E)$ the class of sequences with shifts in $\boldsymbol{\sigma}$ and taking values in the subset of experts $E = \{e_1, \dots, e_n\}$, algorithm GrowingSleepingMarkovHedge with $\pi_i = \frac{1}{\tau_i m_{\tau_i}}$ and $\alpha_t = \beta_t = \frac{1}{t}$ achieves, for every $T \geqslant 1$, $E \subset \{1, \dots, M_T\}$ and $\boldsymbol{\sigma}$,

$$L_T - \inf_{i^T \in \mathscr{S}_T^{(s)}(\boldsymbol{\sigma}, E)} L_T(i^T) \leqslant \frac{1}{\eta} \sum_{p=1}^{n} \left(\log \tau_{e_p} + \log \frac{m_{\tau_{e_p}}}{n} \right) + \frac{1}{\eta} n \log(2T) + \frac{2}{\eta} \sum_{j=1}^{k} \log \sigma_j. \tag{7}$$

In particular, the regret with respect to every admissible sequence of T experts with at most k shifts and taking at most n values is bounded by $\frac{1}{\eta}\left(n \log \frac{\max_{1 \leqslant t \leqslant T} m_t}{n} + 2n \log(\sqrt{2}T) + 2k \log T\right)$.

The main results of this text are Theorem 7, a powerful parameter-free generalization of (Shalizi et al., 2011, Theorem 2), and Theorem 9, which adapts results of Bousquet and Warmuth (2002); Koolen et al. (2012) to sequentially incoming forecasters, and has no precedent in this context.

3. Preliminary: the exponential weights algorithm

First, we introduce the simple but fundamental *exponential weights* or *Hedge algorithm* (Vovk, 1998; Cesa-Bianchi and Lugosi, 2006), designed to control the regret $L_T - L_{i,T} = \sum_{t=1}^T \ell_t - \sum_{t=1}^T \ell_{i,t}$ for a fixed set of experts $\{1, \dots, M\}$. The algorithm depends on a *prior distribution* $\boldsymbol{\pi} \in \mathscr{P}_M$ on the experts and predicts as

$$x_t = \frac{\sum_{i=1}^M w_{i,t}\, x_{i,t}}{\sum_{i=1}^M w_{i,t}} \quad \text{with} \quad w_{i,t} = \pi_i\, e^{-\eta L_{i,t-1}}\,. \tag{8}$$

Equivalently, it forecasts $x_t = \sum_{i=1}^M v_{i,t}\, x_{i,t}$, where the weights $\boldsymbol{v}_t \in \mathscr{P}_M$ are sequentially updated in the following way: $\boldsymbol{v}_1 = \boldsymbol{\pi}$ and, after each round $t \geqslant 1$, $\boldsymbol{v}_{t+1}$ is set to the *posterior* distribution $\boldsymbol{v}_t^m$ of $\boldsymbol{v}_t$ given the losses $(\ell_{i,t})_{1 \leqslant i \leqslant M}$, defined by

$$v_{i,t}^m = \frac{v_{i,t}\, e^{-\eta\, \ell_{i,t}}}{\sum_{j=1}^M v_{j,t}\, e^{-\eta\, \ell_{j,t}}}\,. \tag{9}$$

All subsequent regret bounds will rely on the following standard regret bound (see Appendix A), by reducing complex forecasting strategies to the aggregation of experts under a suitable prior.

Proposition 1 (Cesa-Bianchi and Lugosi (2006, Corollary 3.1)) *Irrespective of the values of the signal and the experts' predictions, the exponential weights algorithm (8) with prior $\boldsymbol{\pi}$ achieves*

$$L_T - L_{i,T} \leqslant \frac{1}{\eta} \log \frac{1}{\pi_i} \tag{10}$$

for each $i = 1, \dots, M$ and $T \geqslant 1$. More generally, for each probability vector $\boldsymbol{u} \in \mathscr{P}_M$,

$$L_T - \sum_{i=1}^M u_i\, L_{i,T} \leqslant \frac{1}{\eta}\, \Delta(\boldsymbol{u} \,\|\, \boldsymbol{\pi})\,. \tag{11}$$

Choosing a uniform prior $\boldsymbol{\pi} = \frac{1}{M}\mathbf{1}$ yields an at most $\frac{1}{\eta} \log M$ regret with respect to the best expert.

4. Growing experts and specialists: the "abstention trick"

A natural idea to tackle the problem of a growing number of experts is to cast it in the related setting of *specialists*, introduced by Freund et al. (1997). We present the specialist setting and the related "specialist trick" identified by Chernov and Vovk (2009) (which we will call the "abstention trick"), which enables to convert any expert aggregation algorithm into a specialist aggregation algorithm. These ideas are then applied to the growing expert ensemble setting, which allows us to control the regret with respect to *constant* experts of equation (2); a refinement is introduced along the way, the use of *unnormalized priors*, that gives more flexibility to the algorithm and its regret bounds.

4.1. Specialists and their aggregation

In the specialist setting, we have access to *specialists* $i \in \{1, \dots, M\}$ that only output predictions at certain steps, while refraining from predicting the rest of the time. In other words, at each step $t \geqslant 1$, only a subset $A_t \subset \{1, \dots, M\}$ of *active* experts output a prediction $x_{i,t} \in \mathscr{X}$.

In order to adapt any expert aggregation strategy to the specialists setting, a crucial idea due to Chernov and Vovk (2009) is to "complete" the specialists' predictions by attributing to inactive specialists $i \notin A_t$ a forecast equal to that of the aggregating algorithm. Although this seems circular, it can be made precise by observing that the only way to simultaneously satisfy the conditions

$$x_t = \sum_{i=1}^{M} v_{i,t}\, x_{i,t} \quad \text{and} \quad x_{i,t} = x_t \ \text{ for any } i \notin A_t \tag{12}$$

is to take

$$x_t = x_{i,t} = \frac{\sum_{i\in A_t} v_{i,t}\, x_{i,t}}{\sum_{i\in A_t} v_{i,t}} \quad \text{for } i \notin A_t\,. \tag{13}$$

We call this technique the "abstention trick", since it consists in attributing to inactive specialists a forecast that will not affect the voting outcome. In the case of the exponential weights algorithm, this leads to the *specialist aggregation* algorithm with prior $\boldsymbol{\pi}$, which forecasts

$$x_t = \frac{\sum_{i\in A_t} w_{i,t}\, x_{i,t}}{\sum_{i\in A_t} w_{i,t}} \quad \text{with} \quad w_{i,t} = \pi_i\, e^{-\eta L_{i,t-1}}\,, \tag{14}$$

where we denote, for each specialist i and $t \geqslant 1$, $L_{i,t} := \sum_{s\leqslant t\,:\, i\in A_s} \ell_{i,s} + \sum_{s\leqslant t\,:\, i\notin A_s} \ell_s$.

Remark 2 *The exp-concavity inequality $e^{-\eta\,\ell_t} \geqslant \sum_{i=1}^{M} v_{i,t}\, e^{-\eta\,\ell_{i,t}}$ shows that $v_{i,t+1} \geqslant v_{i,t}$ for any $i \notin A_t$. In the case of the logarithmic loss, for $\eta = 1$ this inequality becomes an equality, thus the weights of inactive specialists remain unchanged: $v_{i,t+1} = v_{i,t}$.*

Since the specialist aggregation consists of the exponential weights on the extended predictions (13), and since for this extension one has $\sum_{t=1}^{T}(\ell_t - \ell_{i,t}) = \sum_{t\leqslant T\,:\, i\in A_t}(\ell_t - \ell_{i,t})$, Proposition 1 implies:

Proposition 2 (Freund et al. (1997, Theorem 1)) *The specialist aggregation with prior $\boldsymbol{\pi}$ achieves the following regret bound: for each specialist i and every $T \geqslant 1$,*

$$\sum_{t\leqslant T\,:\, i\in A_t} (\ell_t - \ell_{i,t}) \leqslant \frac{1}{\eta}\log\frac{1}{\pi_i}\,. \tag{15}$$

Moreover, for each probability vector $\boldsymbol{u} \in \mathscr{P}_M$, $\displaystyle\sum_{i=1}^{M} u_i \sum_{t\leqslant T\,:\, i\in A_t} (\ell_t - \ell_{i,t}) \leqslant \frac{1}{\eta}\,\Delta(\boldsymbol{u}\,\|\,\boldsymbol{\pi})\,.$

Remark 3 *Note that the sets A_t of active specialists do not need to be known in advance.*

4.2. Adaptation to growing expert ensembles: GrowingHedge

Growing experts can naturally be seen as specialists, by setting $A_t := \{1, \dots, M_t\}$; moreover, through this equivalence, the quantity controlled by Proposition 2 is precisely the regret (2) with respect to *constant experts*. In order to apply the results on specialist aggregation to the growing expert setting, it remains to specify exactly which total set of specialists is considered.

Fixed time horizon. In the simplest case when both the time horizon T and the eventual number of experts M_T are known, the eventual set of experts (at time T) is known, and we can take the finite specialist set to be $\{1, \dots, M_T\}$. Therefore, given any probability vector $\boldsymbol{\pi} = (\pi_1, \dots, \pi_{M_T})$, we can use the aggregation of specialists, with the regret bound (15). In particular, the choice of $\pi_i = \frac{1}{M_T}$ for $i = 1, \dots, M_T$ yields the uniform regret bound $\frac{1}{\eta}\log M_T$.

Anytime algorithm, normalized prior. The fixed horizon approach is somewhat unsatisfactory, since we are typically interested in algorithms that are anytime and agnostic to M_t. To achieve this goal, a better choice is to take the infinite set of specialists $\mathbf{N}^*$. Crucially, the aggregation of this infinite number of specialists can be implemented in finite time, by introducing the weight of an expert *only when it enters*. Given a probability vector $\boldsymbol{\pi} = (\pi_i)_{i\geqslant 1}$ on $\mathbf{N}^*$, this leads to the anytime

strategy GrowingHedge described below. A straightforward adaptation of Propositions 1 and 2 to a countably infinite set of experts shows that this strategy achieves, now for *every* $T \geqslant 1$ and $i \leqslant M_T$, the regret bound (15). However, we are constrained by the fact that $\boldsymbol{\pi}$ must be a probability on $\mathbf{N}^*$.

Anytime algorithm, unnormalized prior. We now turn to the most general analysis, which subsumes and improves the previous two. Now, we let $\boldsymbol{\pi} = (\pi_i)_{i \geqslant 1}$ denote a sequence of *arbitrary* positive weights, that are no longer assumed to sum to 1. These weights do not need to be set in advance: the weight π_i can be chosen when expert i enters, so that at this step τ_i, $(m_t)_{t \leqslant \tau_i}$ and $(M_t)_{t \leqslant \tau_i}$ are known, even if they were unknown at the beginning; in particular, π_i may depend on these quantities. We now consider the anytime algorithm GrowingHedge.

Algorithm 1 GrowingHedge — Anytime aggregation of growing experts

Parameters: Learning rate $\eta > 0$, weights on the experts $\boldsymbol{\pi} = (\pi_i)_{i \geqslant 1}$.
Initialization: Set $w_{i,1} = \pi_i$ for $i = 1, \dots, M_1$.
for $t = 1, 2, \dots$ **do**
  Receive predictions $(x_{1,1}, \dots, x_{M_t,t}) \in \mathscr{X}^{M_t}$ from the experts, and predict

$$x_t = \frac{\sum_{i=1}^{M_t} w_{i,t}\, x_{i,t}}{\sum_{i=1}^{M_t} w_{i,t}}\,. \tag{16}$$

  Observe $y_t \in \mathscr{Y}$, and derive the losses $\ell_t = \ell(x_t, y_t)$ and $\ell_{i,t} = \ell(x_{i,t}, y_t)$.
  Update the weights by $w_{i,t+1} = w_{i,t}\, e^{-\eta\, \ell_{i,t}}$ for $i = 1, \dots, M_t$. Moreover, introduce the weights $w_{i,t+1} = \pi_i\, e^{-\eta\, L_t}$ for $M_t + 1 \leqslant i \leqslant M_{t+1}$.
end for

Theorem 3 *Let $\boldsymbol{\pi} = (\pi_i)_{i \geqslant 1}$ be an arbitrary sequence of positive weights. Then, algorithm GrowingHedge achieves the following regret bound: for every $T \geqslant 1$ and $i \leqslant M_T$,*

$$\sum_{t=\tau_i}^{T} (\ell_t - \ell_{i,t}) \leqslant \frac{1}{\eta} \log\left(\frac{1}{\pi_i} \sum_{j=1}^{M_T} \pi_j\right). \tag{17}$$

Additionally, its time and space complexity at each step $t \geqslant 1$ is $O(M_t)$.

We provide the proof of Theorem 3 in Appendix B. Let us now discuss a few choices of priors, with the corresponding regret bounds (17) (omitting the $\frac{1}{\eta}$ factor).

• With $\pi_i = 1$, we get $\log M_T$, but now with an anytime algorithm. Since $\sum_{i=1}^{M} \frac{1}{i} \leqslant 1 + \sum_{i=2}^{M} \int_{i-1}^{i} \frac{dx}{x} = 1 + \log M$, the choice of $\pi_i = \frac{1}{i}$ yields $\log i + \log(1 + \log M_T)$.

• The above bounds depend on the index $i \geqslant 1$, and hence arbitrarily distinguish experts entered at the same time. More natural bounds would only depend on the entry time τ_i, which is achievable since π_i can be chosen when i enters, and thus depend on τ_i. Setting[2] $\pi_i = \frac{1}{m_{\tau_i}} \nu_{\tau_i}$, where $\boldsymbol{\nu} = (\nu_t)_{t \geqslant 1}$ is a positive sequence set in advance, we get

$$\log m_{\tau_i} + \log \frac{1}{\nu_{\tau_i}} + \log \sum_{t=1}^{T} \nu_t\,. \tag{18}$$

Amongst the many possible choices for ν_t, one may consider $\nu_t = 1$ for which (18) becomes $\log m_{\tau_i} + \log T$, while $\nu_t = \frac{1}{t}$ yields the improved bound $\log m_{\tau_i} + \log \tau_i + \log(1 + \log T)$. Note

2. In fact, this can be slightly refined when $m_t = 0$ for most steps t. In this case, denoting for $t \geqslant 1$: $s(t) = |\{t' \leqslant t \mid m_{t'} \geqslant 1\}|$, we can take $\pi_i = \frac{1}{s(\tau_i)\, m_{\tau_i}}$ and get a regret bound $\frac{1}{\eta} \{\log m_{\tau_i} + \log s(\tau_i) + \log(1 + \log s(T))\}$.

that neither choice is summable, and that a choice of summable weights (e.g. $\nu_t = t^{-\alpha}$, $\alpha > 1$ or $\nu_t = \frac{1}{t\log^2(t+1)}$) generally leads to worse or less interpretable bounds. The first choice ($\nu_t = 1$) is simple, while the second one ($\nu_t = 1/t$) trade-offs simplicity and quality of the bound.

• Another option is to set $\pi_i = \upsilon_{\tau_i}$, where $\boldsymbol{\upsilon} = (\upsilon_t)_{t\geqslant 1}$ is an arbitrary sequence set in advance. The bound becomes

$$\log \frac{1}{\upsilon_{\tau_i}} + \log \sum_{t=1}^{T} m_t \upsilon_t \tag{19}$$

which is more regular than the bound (18) when m_t alternates between small and large values, since it depends on a cumulated quantity instead of just m_{τ_i}. For $\upsilon_t = 1$ (*i.e.* $\pi_i = 1$) this is just $\log M_T$. Alternatively, for $\upsilon_t = \frac{1}{t}$ this becomes $\log \tau_i + \log \sum_{t=1}^{T} \frac{m_t}{t}$.

Regret against sequences of fresh experts. Theorem 3 provides a regret bound against any *static* expert, *i.e.* any constant choice of expert, albeit in a growing experts setting. However, this means that the regret is controlled only on the period $[\![\tau_i, T]\!]$ when the expert actually emits predictions. An alternative way to state Theorem 3 is in terms of *sequences of fresh experts*. Indeed, Theorem 3 implies that, for every sequence of fresh experts i^T with switching times $\sigma_1 < \cdots < \sigma_k$ (with the additional conventions $\sigma_0 := 1$ and $\sigma_{k+1} := T+1$), algorithm GrowingHedge achieves:

$$L_T - L_T(i^T) = \sum_{j=0}^{k} \sum_{t=\sigma_j}^{\sigma_{j+1}-1} (\ell_t - \ell_{i_{\sigma_j}}) \leqslant \frac{1}{\eta} \sum_{j=0}^{k} \log \frac{\Pi_{M_{\sigma_{j+1}-1}}}{\pi_{i_{\sigma_j}}} \tag{20}$$

since $\sigma_j = \tau_{i_{\sigma_j}}$, and where we denote $\Pi_M = \sum_{i=1}^{M} \pi_i$ for each $M \geqslant 1$. Taking $\pi_i = 1$, this bound reduces to $\frac{1}{\eta}\sum_{j=0}^{k} \log M_{\sigma_{j+1}-1} \leqslant \frac{1}{\eta}(k+1)\log M_T$. Taking $\pi_i = 1/m_{\tau_i}$, so that $\Pi_{M_t} = t$, and further bounding $\Pi_{M_{\sigma_{j+1}-1}} = \sigma_{j+1} - 1 \leqslant \sigma_{j+1}$ for $0 \leqslant j \leqslant k-1$ and $\Pi_{M_{\sigma_{k+1}-1}} = T$, we recover the bound (5) stated in the overview.

5. Growing experts and sequences of experts: the "muting trick"

Algorithm GrowingHedge, based on the specialist viewpoint, guarantees good regret bounds against *fresh* sequences of experts and admits an efficient implementation. Instead of comparing only against fresh sequences of experts, it may be preferable to target *arbitrary* admissible sequences of experts, that contain transitions to incumbent experts; this could be beneficial when some experts start predicting well after a few rounds. A natural approach consists in applying the abstention trick to algorithms for a fixed expert set that target arbitrary sequences of experts (such as Fixed Share, see Appendix C). As it turns out, such an approach would require to maintain weights for unentered experts (which may be in unknown, even infinite, number in an anytime setting): the fact that one could obtain an efficient algorithm such as GrowingHedge was specific to the exponential weights algorithm, and does not extend to more sophisticated algorithms that perform weight sharing.

In this section, we adopt a "dual" point of view, which proves more flexible. Indeed, in the growing expert ensemble setting, there are two ways to cope with the fact that some experts' predictions are undefined at each step. The abstention trick amounts to attributing *predictions* to the experts which have not entered yet, so that they do not affect the learner's forecast. Another option is to design a prior on *sequences* of experts so that the *weight* of unentered experts is 0, and hence their predictions are irrelevant[3]; we call this the "muting trick".

3. In this case, the learner's predictions do not depend on the way we complete the experts' predictions, so the algorithm may be defined even when experts with zero weight do not output predictions.

After reviewing the well-known setting of aggregation of sequences of experts for a fixed set of experts (Section 5.1) and presenting the generic algorithm MarkovHedge with its regret bound, we adapt it to the growing experts setting by providing FreshMarkovHedge (Section 5.2) and Growing-MarkovHedge (Section 5.3), that compete respectively with fresh and arbitrary sequences.

5.1. Aggregating sequences of experts

The problem of controlling the regret with respect to sequences of experts, known as *tracking the best expert*, was introduced by Herbster and Warmuth (1998), who proposed the simple *Fixed Share* algorithm with good regret guarantees. A key fact, first recognized by Vovk (1999), is that Fixed Share, and in fact many other weight sharing algorithms (Koolen and de Rooij, 2008, 2013), can be interpreted as the exponential weights on sequences of experts under a suitable prior. We will state this result in the general form of Lemma 4, which implies the regret bound of Proposition 5.

Markov prior. If $i^T = (i_1, \dots, i_T)$ is a finite sequence of experts, its predictions up to time T are derived from those of the base experts $i \in \{1, \dots, M\}$ in the following way: $x_t(i^T) = x_{i_t,t}$ for $1 \leqslant t \leqslant T$. Given a prior distribution $\pi = (\pi(i^T))_{i^T}$, we could in principle consider the exponentially weighted aggregation of sequences under this prior; however, such an algorithm would be intractable even for moderately low values of T, since it would require to store and update $O(M^T)$ weights. Fortunately, when $\pi(i_1, \dots, i_T) = \theta_1(i_1)\,\theta_2(i_2 \,|\, i_1) \cdots \theta_T(i_T \,|\, i_{T-1})$ is a Markov probability distribution with initial measure $\boldsymbol{\theta}_1$ and transition matrices $\boldsymbol{\theta}_t$, $2 \leqslant t \leqslant T$, the exponentially weighted aggregation under the prior π collapses to the efficient algorithm MarkovHedge.

Algorithm 2 MarkovHedge — Aggregation of sequences of experts under a Markov prior

1: **Parameters:** Learning rate $\eta > 0$, initial weights $\boldsymbol{\theta}_1 = (\theta_1(i))_{1 \leqslant i \leqslant M}$, and transition probabilities $\boldsymbol{\theta}_t = \big(\theta_t(i \,|\, j)\big)_{1 \leqslant i,j \leqslant M}$ for all $t \geqslant 2$.
2: **Initialization:** Set $\boldsymbol{v}_1 = \boldsymbol{\theta}_1$.
3: **for** $t = 1, 2, \dots$ **do**
4: Receive predictions $\boldsymbol{x}_t \in \mathscr{X}^M$ from the experts, and predict $x_t = \boldsymbol{v}_t \cdot \boldsymbol{x}_t$.
5: Observe $y_t \in \mathscr{Y}$, then derive the losses $\ell_t = \ell(x_t, y_t)$ and $\ell_{i,t} = \ell(x_{i,t}, y_t)$ and the posteriors

$$v_{i,t}^m = \frac{v_{i,t}\, e^{-\eta\, \ell_{i,t}}}{\sum_{j=1}^M v_{j,t}\, e^{-\eta\, \ell_{j,t}}}\,. \tag{21}$$

6: Update the weights by $\boldsymbol{v}_{t+1} = \boldsymbol{\theta}_{t+1}\, \boldsymbol{v}_t^m$, *i.e.*

$$v_{i,t+1} = \sum_{j=1}^M \theta_{t+1}(i \,|\, j)\, v_{j,t}^m\,. \tag{22}$$

7: **end for**

Remark 4 *Algorithm MarkovHedge only requires to store and update $O(M)$ weights. Due to the matrix product (22), the update may take a $O(M^2)$ time; however, all the transition matrices we consider lead to a simple update in $O(M)$ time.*

Lemma 4 *For every $T \geqslant 1$, the forecasts of algorithm MarkovHedge coincide up to time T with those of the exponential aggregation of finite sequences of experts $i^T = (i_1, \dots, i_T)$ under the Markov prior with initial distribution $\boldsymbol{\theta}_1$ and transition matrices $\boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_T$.*

Lemma 4 – proven in Appendix C – and Proposition 1 directly imply the following regret bound.

Proposition 5 *Algorithm MarkovHedge, with initial distribution $\boldsymbol{\theta}_1$ and transition matrices $\boldsymbol{\theta}_t$, guarantees the following regret bound: for every $T \geqslant 1$ and any sequence of experts $(i_1, \dots, i_T)$,*

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_{i_t,t} \leqslant \frac{1}{\eta} \log \frac{1}{\theta_1(i_1)} + \frac{1}{\eta} \sum_{t=2}^{T} \log \frac{1}{\theta_t(i_t \,|\, i_{t-1})} \,. \tag{23}$$

It is worth noting that the transition probabilities $\boldsymbol{\theta}_t$ only intervene at step t in algorithm MarkovHedge, and hence they can be chosen at this time.

Notable examples. In Appendix C, we discuss particular instances of MarkovHedge that lead to well-known algorithms (such as Fixed Share), and recover their regret bounds using Proposition 5.

5.2. Application to sequences of fresh experts

We now explain how to specify the generic algorithm MarkovHedge in order to adapt it to the growing experts setting. This adaptation relies on the "muting trick": to obtain a strategy which is well-defined for growing experts, one has to ensure that experts who do not predict have zero weight, which amounts to saying that all weight is put to *admissible* sequences of experts. Importantly, this is possible even when the numbers M_t are not known from the beginning, since *the transition matrices $\boldsymbol{\theta}_t$ can be chosen at time t*, when M_t is revealed.

We start in this section by designing an algorithm FreshMarkovHedge that compares to sequences of fresh experts; to achieve this, it is natural to design a prior that assigns full probability to sequences of fresh experts. It turns out that we can recover an algorithm similar to the algorithm GrowingHedge, with the same regret guarantees, through this seemingly different viewpoint.

Let $\boldsymbol{\pi} = (\pi_i)_{i \geqslant 1}$ be an *unnormalized prior* as in Section 4.2. For each $M \geqslant 1$, we denote $\Pi_M = \sum_{i=1}^{M} \pi_i$. We consider the following transition matrices $\boldsymbol{\theta}_t$ in strategy MarkovHedge:

$$\theta_1(i) = \frac{\pi_i}{\Pi_{M_1}} \mathbf{1}_{i \leqslant M_1} \quad ; \quad \theta_{t+1}(i \,|\, j) = \frac{\Pi_{M_t}}{\Pi_{M_{t+1}}} \mathbf{1}_{i=j} + \frac{\pi_i}{\Pi_{M_{t+1}}} \mathbf{1}_{M_t+1 \leqslant i \leqslant M_{t+1}} \tag{24}$$

for every $i \geqslant 1$, $t \geqslant 1$ and $j \in \{1, \dots, M_t\}$. The other transition probabilities $\theta_{t+1}(i \,|\, j)$ for $j > M_t$ are irrelevant; indeed, a simple induction shows that $v_{j,t} = 0$ for every $j > M_t$, so that the instantiation of algorithm MarkovHedge with the transition probabilities (24) leads to the forecasts

$$x_t = \sum_{i=1}^{M_t} v_{i,t} \, x_{i,t} \tag{25}$$

(which do not depend on the undefined prediction of the experts $i > M_t$) where the weights $(v_{i,t})_{1 \leqslant i \leqslant M_t}$ are recursively defined by $v_{i,1} = \frac{\pi_i}{\Pi_{M_1}}$ $(1 \leqslant i \leqslant M_1)$ and the update

$$v_{i,t+1} = \frac{\Pi_{M_t}}{\Pi_{M_{t+1}}} \, v_{i,t}^m \quad (1 \leqslant i \leqslant M_t) \,; \quad v_{i,t+1} = \frac{\pi_i}{\Pi_{M_{t+1}}} \quad (M_t + 1 \leqslant i \leqslant M_{t+1}) \,, \tag{26}$$

where we set $v_{i,t}^m = \dfrac{v_{i,t} \, e^{-\eta \, \ell_{i,t}}}{\sum_{j=1}^{M_t} v_{j,t} \, e^{-\eta \, \ell_{j,t}}}$ for $1 \leqslant i \leqslant M_t$. We call this algorithm FreshMarkovHedge.

Theorem 6 *Algorithm FreshMarkovHedge using weights $\boldsymbol{\pi}$ achieves the following regret bound: for every $T \geqslant 1$ and sequence of fresh experts $i^T = (i_1, \dots, i_T)$ with shifts at times $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_k)$,*

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} \sum_{j=0}^{k} \log \frac{1}{\pi_{i_{\sigma_j}}} + \frac{1}{\eta} \sum_{j=1}^{k} \log \Pi_{M_{\sigma_j - 1}} + \frac{1}{\eta} \log \Pi_{M_T} \,. \tag{27}$$

Additionally, the time and space complexity of the algorithm at each time step $t \geqslant 1$ is $O(M_t)$.

Proof For any sequence of fresh experts $i^T \in \mathscr{S}_T^{(f)}(\boldsymbol{\sigma})$, replacing in the bound (23) of Proposition 5 the conditional probabilities $\theta_{t+1}(i_{t+1} \,|\, i_t)$ by their values (defined by (24)), we get

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} \sum_{j=0}^{k} \left\{ \log\left(\frac{1}{\pi_{i_{\sigma_j}}} \Pi_{M_{\sigma_j}} \right) + \sum_{t=\sigma_j+1}^{\sigma_{j+1}-1} \log \frac{\Pi_{M_t}}{\Pi_{M_{t-1}}} \right\} = \frac{1}{\eta} \sum_{j=0}^{k} \log \frac{\Pi_{M_{\sigma_{j+1}-1}}}{\pi_{i_{\sigma_j}}}$$

which is precisely the desired bound (27). ■

Remark 5 *The regret bound (27) of the FreshMarkovHedge algorithm against sequences of fresh experts is exactly the same as the one of the GrowingHedge algorithm (20). This is not a coincidence: the two algorithms are almost identical, except that expert i is introduced with a weight $\pi_i / (\sum_{i=1}^{M_{\tau_i}} \pi_i)$ by FreshMarkovHedge and $\pi_i e^{-\eta L_{\tau_i - 1}} / (\sum_{j=1}^{M_{\tau_i}} \pi_j e^{-\eta L_{j, \tau_i - 1}})$ by GrowingHedge. In the case of the logarithmic loss (with $\eta = 1$), these two weights are equal (see Remark 2), and hence the strategies GrowingHedge and FreshMarkovHedge coincide.*

5.3. Regret against arbitrary sequences of experts

We now consider the more ambitious objective of comparing to *arbitrary* admissible sequences of experts. This can be done by using another choice of transition matrices, which puts all the weight to admissible sequences of experts (and not just sequences of fresh experts).

Algorithm GrowingMarkovHedge instantiates MarkovHedge on the transition matrices

$$\theta_1(i) = \frac{\pi_i}{\Pi_{M_1}} \mathbf{1}_{i \leqslant M_1} \quad ; \quad \theta_{t+1}(i \,|\, j) = \alpha_{t+1} \frac{\pi_i}{\Pi_{M_{t+1}}} + (1 - \alpha_{t+1})\, \theta_{t+1}^{(f)}(i \,|\, j) \tag{28}$$

where $\boldsymbol{\theta}_t^{(f)}$ denote the transition matrices of algorithm FreshMarkovHedge. As before, this leads to a well-defined growing experts algorithm which predicts $x_t = \sum_{i=1}^{M_t} v_{i,t}\, x_{i,t}$, where the weights $(v_{i,t})_{1 \leqslant i \leqslant M_t}$ are recursively defined by $v_{i,1} = \frac{\pi_i}{\Pi_{M_1}}$ $(1 \leqslant i \leqslant M_1)$ and the update

$$\begin{gathered} v_{i,t+1} = (1-\alpha_{t+1}) \frac{\Pi_{M_t}}{\Pi_{M_{t+1}}} v_{i,t}^m + \alpha_{t+1} \frac{\pi_i}{\Pi_{M_{t+1}}} \; (1 \leqslant i \leqslant M_t) \,; \; v_{i,t+1} = \frac{\pi_i}{\Pi_{M_{t+1}}} \; (M_t+1 \leqslant i \leqslant M_{t+1}) \,, \\ \text{where again } v_{i,t}^m = \frac{v_{i,t}\, e^{-\eta \ell_{i,t}}}{\sum_{j=1}^{M_t} v_{j,t}\, e^{-\eta \ell_{j,t}}} \text{ for } 1 \leqslant i \leqslant M_t. \text{ In this case, Proposition 5 yields:} \end{gathered} \tag{29}$$

Theorem 7 *Algorithm GrowingMarkovHedge based on the weights $\boldsymbol{\pi}$ and parameters $(\alpha_t)_{t \geqslant 2}$ achieves the following regret bound: for every $T \geqslant 1$, and every admissible sequence of experts $i^T = (i_1, \ldots, i_T)$ with shifts at times $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_k)$,*

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} \left\{ \sum_{j=0}^{k} \log \frac{\Pi_{M_{\sigma_{j+1}-1}}}{\pi_{i_{\sigma_j}}} + \sum_{j=1}^{k_1} \log \frac{1}{\alpha_{\sigma_j^1}} + \sum_{2 \leqslant t \leqslant T \,:\, t \notin \boldsymbol{\sigma}} \log \frac{1}{1 - \alpha_t} \right\}. \tag{30}$$

where $\boldsymbol{\sigma}^0 = (\sigma_1^0, \ldots, \sigma_{k_0}^0)$ (resp. $\boldsymbol{\sigma}^1 = (\sigma_1^1, \ldots, \sigma_{k_1}^1)$) denotes the shifts to fresh (resp. incumbent) experts, with $k = k_0 + k_1$. Moreover, it has a $O(M_t)$ time and space complexity at each step $t \geqslant 1$.

Remark 6 *Note that by choosing $\alpha_t = \frac{1}{t}$, we have, since $\frac{1}{1-1/t} = \frac{t}{t-1}$,*

$$\sum_{j=1}^{k_1} \log \frac{1}{\alpha_{\sigma_j^1}} + \sum_{2 \leqslant t \leqslant T \,:\, t \notin \boldsymbol{\sigma}} \log \frac{1}{1 - \alpha_t} \leqslant \sum_{j=1}^{k_1} \log \sigma_j^1 + \sum_{t=2}^{T} \log \frac{t}{t-1} = \sum_{j=1}^{k_1} \log \sigma_j^1 + \log T \,.$$

Additionally, by setting $\pi_i = 1$ *the bound (30) becomes* $\frac{1}{\eta}(\sum_{j=0}^{k} \log M_{\sigma_{j+1}-1} + \sum_{j=1}^{k_1} \log \sigma_j^1 + \log T)$*, which is lower than* $\frac{1}{\eta}(k+1)\log M_T + \frac{1}{\eta}(k_1+1)\log T$*. We can also recover the bound (6) by setting* $\pi_i = \frac{1}{\tau_i m_{\tau_i}}$*, since in this case we have* $\Pi_{M_{\sigma_{j+1}-1}} \leqslant \Pi_{M_T} \leqslant \sum_{t=1}^{T} \frac{1}{t} \leqslant 1 + \log T$*.*

6. Combining growing experts and sequences of sleeping experts

Sections 4 and 5 studied the problem of growing experts using tools from two different settings (specialists and sequences of experts). Drawing on ideas from Koolen et al. (2012), we show in this section how to combine these two frameworks, in order to address the more challenging problem of controlling the regret with respect to *sparse sequences of experts* in the growing experts setting. Note that the refinement to sparse sequences of experts is particularly relevant in the context of a growing experts ensemble, since in this context the total number of experts will typically be large.

6.1. Sleeping experts: generic result

The problem of comparing to sparse sequences of experts, or *tracking a small pool of experts*, is a refinement on the problem of tracking the best expert. The seminal paper (Bousquet and Warmuth, 2002) proposed an ad-hoc strategy with essentially optimal regret bounds, the *Mixing Past Posteriors* (MPP) algorithm (see also Cesa-Bianchi et al. (2012)). A full "bayesian" interpretation of this algorithm in terms of the aggregation of "sleeping experts" was given by Koolen et al. (2012), which enabled the authors to propose a more efficient alternative. Here, by reinterpreting this construction, we propose a more general algorithm and regret bound (Proposition 8); this extension will be crucial to adapt this strategy to the growing experts setting (Section 6.2).

Given a fixed set of experts $\{1, \dots, M\}$, we call *sleeping expert* a couple $(i, a) \in \{1, \dots, M\} \times \{0, 1\}$; we endow the set of sleeping experts with a specialist structure by deciding that (i, a) is active if and only if $a = 1$, and that $x_t(i, 1) := x_{i,t}$ is the prediction of expert i. A key insight from Koolen et al. (2012) is to decompose the regret with respect to a sparse sequence $i^T = (i_1, \dots, i_T)$ of experts, taking values in the set $\{e_p \mid 1 \leqslant p \leqslant n\}$, in the following way:

$$\sum_{t=1}^{T}(\ell_t - \ell_{i_t,t}) = \sum_{p=1}^{n} \sum_{t \leqslant T\,:\, i_t = e_p} (\ell_t - \ell_{e_p,t}) = \sum_{p=1}^{n}\sum_{t=1}^{T}(\ell_t - \ell_t(e_p, a_{p,t})) = n \sum_{\iota^T} u(\iota^T)(L_T - L_T(\iota^T))$$

where $a_{p,t} := \mathbf{1}_{i_t = e_p}$, and u is the probability distribution on the sequences ι^T of sleeping experts which is uniform on the n sequences $\iota_p^T = (e_p, a_{p,t})_{1 \leqslant t \leqslant T}$, $p = 1, \dots, n$. Note that in the second equality we used the "abstention trick", which attributes to inactive sleeping experts $(e_p, 0)$ the prediction x_t of the algorithm.

We can now aggregate sequences of sleeping experts under a Markov prior, given initial weights $\theta_1(i, a)$ and transition probabilities $\theta_{t+1}(i_{t+1}, a_{t+1} \mid i_t, a_t)$, recalling that $\boldsymbol{\theta}_t$ can be chosen at step t. For convenience, we restrict here to transitions that only occur between sleeping experts (i, a) with the same base expert, and denote $\theta_{i,t}(a \mid b) = \theta_t(i, a \mid i, b)$ for $a, b \in \{0, 1\}$. This leads to the algorithm SleepingMarkovHedge.

Remark 7 *The structure of our prior is slightly more general than the one used by Koolen et al. (2012), which considered priors on couples* (i, a^T) *with an independence structure:* $\pi(i, a^T) = \pi(i)\,\pi(a^T)$*, with* $\pi(a^T)$ *a Markov distribution, which amounts to saying that the transition probabilities* $\theta_{i,t}(a \mid b)$ *could not depend on* i*. This additional flexibility will enable in Section 6.2 the "muting trick", which allows to convert SleepingMarkovHedge to the growing experts setting.*

Additionally, allowing transitions between sleeping experts $(i,1)$ and $(j,1)$ for $i \neq j$ may be interesting in its own right, e.g. if one seeks to control at the same time *the regret with respect to sparse and non-sparse sequences of experts.*

Algorithm 3 SleepingMarkovHedge: sequences of sleeping experts under a Markov chain prior

1: **Parameters:** Learning rate $\eta > 0$, (normalized) prior $\boldsymbol{\pi}$ on the experts, initial wake/sleep probabilities $\theta_{i,1}(a)$, transition probabilities $\boldsymbol{\theta}_{i,t} = \big(\theta_{i,t}(a \,|\, b)\big)_{a,b\in\{0,1\}}$ for $t \geqslant 2$, $1 \leqslant i \leqslant M$.

2: **Initialization:** Set $v_1(i,a) = \pi_i\, \theta_{i,1}(a)$ for $i = 1, \dots, M$ and $a \in \{0,1\}$.

3: **for** $t = 1, 2, \dots$ **do**

4: Receive predictions $\boldsymbol{x}_t \in \mathscr{X}^M$ from the experts, and predict

$$x_t = \frac{\sum_{i=1}^M v_t(i,1)\, x_{i,t}}{\sum_{i=1}^M v_t(i,1)}\,. \tag{31}$$

5: Observe $y_t \in \mathscr{Y}$, then derive the losses $\ell_t(i,0) = \ell_t = \ell(x_t, y_t)$, $\ell_t(i,1) = \ell_{i,t} = \ell(x_{i,t}, y_t)$ and the posteriors

$$v_t^m(i,a) = \frac{v_t(i,a)\, e^{-\eta\, \ell_t(i,a)}}{\sum_{i',a'} v_t(i',a')\, e^{-\eta\, \ell_t(i',a')}}\,. \tag{32}$$

6: Update the weights by

$$v_{t+1}(i,a) = \sum_{b\in\{0,1\}} \theta_{i,t+1}(a \,|\, b)\, v_t^m(i,b)\,. \tag{33}$$

7: **end for**

Proposition 8 *Strategy SleepingMarkovHedge guarantees the following regret bound: for each sequence i^T of experts taking values in the pool $\{e_p \mid 1 \leqslant p \leqslant n\}$, denoting $a_{p,t} = \mathbf{1}_{i_t = e_p}$*

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} \sum_{p=1}^n \left(\log \frac{1/n}{\pi_{e_p}} + \log \frac{1}{\theta_{e_p,1}(a_{p,1})} + \sum_{t=2}^T \log \frac{1}{\theta_{e_p,t}(a_{p,t} \,|\, a_{p,t-1})} \right). \tag{34}$$

The proof of Proposition 8 is given in Appendix D.

6.2. Sparse shifting regret for growing experts

We show here how to instantiate algorithm SleepingMarkovHedge in order to adapt it to the growing experts setting. Again, we use a "muting trick" which attributes a zero weight to experts that have not entered.

Let us consider prior weights $\boldsymbol{\pi} = (\pi_i)_{i\geqslant 1}$ on the experts, which may be unnormalized and chosen at entry time. Let $\alpha_t, \beta_t \in (0,1)$ for $t \geqslant 2$. We set $\theta_{i,1}(1) = \frac{1}{2}$ for $i = 1, \dots, M_1$ and 0 otherwise; moreover, for every $t \geqslant 1$, we take $\theta_{i,t+1}(1 \,|\, \cdot) = 0$ for $i > M_{t+1}$ (recall that $\boldsymbol{\theta}_{i,t+1}$ can be chosen at step $t+1$), $\theta_{i,t+1}(1 \,|\, \cdot) = \frac{1}{2}$ if $M_t + 1 \leqslant i \leqslant M_{t+1}$, and for $i \leqslant M_t$: $\theta_{i,t+1}(0 \,|\, 1) = \alpha_{t+1}$, $\theta_{i,t+1}(1 \,|\, 0) = \beta_{t+1}$. The algorithm obtained with these choices, which we call GrowingSleepingMarkovHedge, is well-defined and predicts $x_t = (\sum_{i=1}^{M_t} v_t(i,1)\, x_{i,t})/(\sum_{i=1}^{M_t} v_t(i,1))$, where the weights $(v_t(i,a))_{1\leqslant i\leqslant M_t,\, a\in\{0,1\}}$ are defined by $v_1(i,a) = \frac{1}{2}\,\pi_i$ $(1 \leqslant i \leqslant M_1)$ and by the update

$$v_{t+1}(i,a) = \sum_{b\in\{0,1\}} \theta_{i,t+1}(a \,|\, b)\, v_t^m(i,b)\ (1 \leqslant i \leqslant M_t)\,;\ v_{t+1}(i,a) = \frac{1}{2}\,\pi_i\ (M_t + 1 \leqslant i \leqslant M_{t+1})\,,$$

with $v_t^m(i,a) = v_t(i,a)\, e^{-\eta\, \ell_t(i,a)} / \sum_{i=1}^{M_t} \sum_{a'\in\{0,1\}} v_t(i',a')\, e^{-\eta\, \ell_t(i',a')}$ for $1 \leqslant i \leqslant M_t$.

Theorem 9 *Algorithm GrowingSleepingMarkovHedge guarantees the following: for each $T \geqslant 1$ and any sequence i^T of experts taking values in the pool $\{e_p \mid 1 \leqslant p \leqslant n\}$, denoting $a_{p,t} = \mathbf{1}_{i_t = e_p}$*

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} \sum_{p=1}^{n} \log \frac{\Pi_{M_T}/n}{\pi_{e_p}} + \frac{1}{\eta} n \log 2 + \frac{1}{\eta} \sum_{t=2}^{T} \left[\log \frac{1}{1-\alpha_t} + (n-1) \log \frac{1}{1-\beta_t} \right] + \frac{1}{\eta} \sum_{j=1}^{k} \left(\log \frac{1}{\alpha_{\sigma_j}} + \log \frac{1}{\beta_{\sigma_j}} \right) \tag{35}$$

where $\boldsymbol{\sigma} = \sigma_1 < \cdots < \sigma_k$ denote the shifting times of i^T. Moreover, the algorithm has a $O(M_t)$ time and space complexity at step t, for every $t \geqslant 1$.

In particular, Theorem 9 enables to recover the bound (7) for $\alpha_t = \beta_t = \frac{1}{t}$ and $\pi_i = \frac{1}{\tau_i m_{\tau_i}}$.

Proof Note that algorithm GrowingSleepingMarkovHedge is invariant under any change of prior $\boldsymbol{\pi} \leftarrow \lambda \boldsymbol{\pi}$ due to the renormalisation in the formula defining x_t. In particular, setting $\lambda = 1/\Pi_{M_T}$, we see that it coincides up to time T with algorithm SleepingMarkovHedge with set of experts $\{1, \ldots, M_T\}$ and (normalized) prior weights π_i / Π_{M_T}. The bound (35) is now a consequence of the general regret bound (34), by substituting for the values of $\boldsymbol{\theta}_{i,t+1}$. ■

Conclusion. In this paper, we extended aggregation of experts to the *growing expert* setting, where novel experts are made available at any time. In this context when the set of experts itself varies, it is natural to seek to track the best expert; different comparison classes of increasing complexity were considered. In order to obtain efficient algorithms with a per-round complexity linear in the current number of experts, we started with generic reformulation of existing algorithms for fixed expert set, and identified two orthogonal techniques (the "abstention trick" from the specialist literature, and the "muting trick") to adapt them to sequentially incoming forecasters. Combined with a proper tuning of the parameters of the prior, this enabled us to obtain tight regret bounds, adaptive to the parameters of the comparison class. Along the way, we recovered several key results from the literature as special case of our analysis, in a somewhat unified approach.

Although we considered the exp-concave assumption to avoid distracting the reader from the main challenges of the growing expert setting, extending our results to the bounded convex case in which the parameter η needs to be adaptively tuned seems possible and is left for future work. In addition, building on the recent work of Jun et al. (2017) might bring further improvements in this case. Another natural extension of our work would be to address the same questions in the framework of online convex optimization (Shalev-Shwartz, 2012; Hazan, 2016), when the gradient of the loss function is made available at each time step.

Acknowledgments

This work has been supported by the French Agence Nationale de la Recherche (ANR), under grant ANR-16- CE40-0002 (project BADASS), and Inria. JM acknowledges support from École Polytechnique fund raising – Data Science Initiative.

References

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

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

Nicolò Cesa-Bianchi, Pierre Gaillard, Gábor Lugosi, and Gilles Stoltz. Mirror descent meets fixed share (and feels no regret). In *Advances in Neural Information Processing Systems 25*, pages 980–988. Curran Associates, Inc., 2012.

Alexey Chernov and Vladimir Vovk. Prediction with expert evaluators' advice. In *Proceedings of the 20th international conference on Algorithmic learning theory*, ALT '09, pages 8–22, Berlin, Heidelberg, 2009. Springer-Verlag.

Steven de Rooij, Tim van Erven, Peter Grünwald, and Wouter M. Koolen. Follow the leader if you can, hedge if you must. *Journal of Machine Learning Research*, 15:1281–1316, 2014.

Yoav Freund, Robert E. Schapire, Yoram Singer, and Manfred K. Warmuth. Using and combining predictors that specialize. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC)*, pages 334–343, 1997.

Eyal Gofer, Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Regret minimization for branching experts. In *Proceedings of the 26th Annual Conference on Learning Theory (COLT)*, pages 618–638, 2013.

László Györfi, Gábor Lugosi, and Gustáv Morvai. A simple randomized algorithm for sequential prediction of ergodic time series. *IEEE Transactions on Information Theory*, 45(7):2642–2650, 1999.

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

David Haussler, Jyrki Kivinen, and Manfred K. Warmuth. Sequential prediction of individual sequences under general loss functions. *IEEE Transactions on Information Theory*, 44(5):1906–1925, 1998.

Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3-4):157–325, 2016.

Elad Hazan and Comandur Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th annual international conference on machine learning*, ICML '09, pages 393–400, 2009.

Mark Herbster and Manfred K. Warmuth. Tracking the best expert. *Machine Learning*, 32(2): 151–178, August 1998.

Kwang-Sung Jun, Francesco Orabona, Stephen Wright, and Rebecca Willett. Improved Strongly Adaptive Online Learning using Coin Betting. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54, pages 943–951, 2017.

Wouter M. Koolen and Steven de Rooij. Combining expert advice efficiently. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT)*, 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.

Wouter M. Koolen and Tim van Erven. Second-order quantile methods for experts and combinatorial games. In *Proceedings of the 28th Annual Conference on Learning Theory (COLT)*, pages 1155–75, 2015.

Wouter M. Koolen, Dmitry Adamskiy, and Manfred K. Warmuth. Putting bayes to sleep. In *Advances in Neural Information Processing Systems 25*, pages 135–143. Curran Associates, Inc., 2012.

Wouter M. Koolen, Tim van Erven, and Peter Grünwald. Learning the learning rate for prediction with expert advice. In *Advances in Neural Information Processing Systems 27*, pages 2294–2302. Curran Associates, Inc., 2014.

Haipeng Luo and Robert E. Schapire. Achieving all with no parameters: Adaptive normalhedge. In *Proceedings of the 28th Annual Conference on Learning Theory (COLT)*, pages 1286–1304, 2015.

Scott McQuade and Claire Monteleoni. Global climate model tracking using geospatial neighborhoods. In *AAAI*, 2012.

Neri Merhav and Meir Feder. Universal prediction. *IEEE Transactions on Information Theory*, 44: 2124–2147, 1998.

Claire Monteleoni, Gavin A Schmidt, Shailesh Saroha, and Eva Asplund. Tracking climate models. *Statistical Analysis and Data Mining*, 4(4):372–392, 2011.

Boris Y. Ryabko. Twice-universal coding. *Problems of information transmission*, 20(3):173–177, 1984.

Boris Y. Ryabko. Prediction of random sequences and universal coding. *Problems of information transmission*, 24(2):87–96, 1988.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Found. Trends Mach. Learn.*, 4(2):107–194, February 2012.

Cosma Rohilla Shalizi, Abigail Z. Jacobs, Kristina Lisa Klinkner, and Aaron Clauset. Adapting to non-stationarity with growing expert ensembles. *arXiv preprint* `arXiv:1103.0949`, 2011.

Gil Shamir and Neri Merhav. Low-complexity sequential lossless coding for piecewise-stationary memoryless sources. *IEEE transactions on information theory*, 45(5):1498–1519, 1999.

Gilles Stoltz. Agrégation séquentielle de prédicteurs : méthodologie générale et applications à la prévision de la qualité de l'air et à celle de la consommation électrique. *Journal de la Société Française de Statistique*, 151(2):66–106, 2010.

Vladimir Vovk. A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56(2):153–173, 1998.

Vladimir Vovk. Derandomizing stochastic prediction strategies. *Machine Learning*, 35(3):247–282, 1999.

Frans M. J. Willems. Coding for a binary independent piecewise-identically-distributed source. *IEEE transactions on information theory*, 42(6):2210–2217, 1996.

Olivier Wintenberger. Optimal learning with Bernstein online aggregation. *Machine Learning*, 106 (1):119–141, 2017.

Appendix A. Proof of Proposition 1

Proof Since the loss function is η-exp-concave and $x_t = \sum_{i=1}^M v_{i,t}\, x_{i,t}$, we have

$$e^{-\eta\,\ell(x_t,y_t)} \geqslant \sum_{i=1}^M v_{i,t}\, e^{-\eta\,\ell(x_{i,t},y_t)}, \qquad \textit{i.e.} \qquad \ell_t \leqslant -\frac{1}{\eta}\log\left(\sum_{i=1}^M v_{i,t}\, e^{-\eta\,\ell_{i,t}}\right).$$

This yields, introducing the posterior weights $v_{i,t}^m$ defined by (9),

$$\ell_t - \ell_{i,t} \leqslant -\frac{1}{\eta}\log\left(\sum_{j=1}^M v_{j,t}\, e^{-\eta\,\ell_{j,t}}\right) - \ell_{i,t} = \frac{1}{\eta}\log\left(\frac{e^{-\eta\,\ell_{i,t}}}{\sum_{j=1}^M v_{j,t}\, e^{-\eta\,\ell_{j,t}}}\right) = \frac{1}{\eta}\log\frac{v_{i,t}^m}{v_{i,t}}\,.$$

Now recalling that the exponentially weighted average forecaster uses $\boldsymbol{v}_{t+1} = \boldsymbol{v}_t^m$, this writes: $\ell_t - \ell_{i,t} \leqslant \frac{1}{\eta}\log\frac{v_{i,t+1}}{v_{i,t}}$ which, summing over $t = 1,\dots,T$, yields $L_T - L_{i,T} \leqslant \frac{1}{\eta}\log\frac{v_{i,T+1}}{v_{i,1}}$. Since $v_{i,1} = \pi_i$ and $v_{i,T+1} \leqslant 1$, this proves (10); moreover, noting that $\log\frac{v_{i,T+1}}{v_{i,1}} = \log\frac{u_i}{v_{i,1}} - \log\frac{u_i}{v_{i,T+1}}$, this implies

$$\sum_{i=1}^M u_i\,(L_T - L_{i,T}) \leqslant \frac{1}{\eta}\sum_{i=1}^M u_i\,\log\frac{v_{i,T+1}}{v_{i,1}} = \frac{1}{\eta}\big(\Delta(\boldsymbol{u}\,\|\,\boldsymbol{v}_1) - \Delta(\boldsymbol{u}\,\|\,\boldsymbol{v}_{T+1})\big)\,,$$

which establishes (11) since $\boldsymbol{v}_1 = \boldsymbol{\pi}$ and $\Delta(\boldsymbol{u}\,\|\,\boldsymbol{v}_{T+1}) \geqslant 0$. ■

Remark 8 *We can recover the bound (10) from inequality (11) by considering $\boldsymbol{u} = \delta_i$. Conversely, inequality (10) implies, by convex combination,*

$$L_T - \sum_{i=1}^M u_i\, L_{i,T} \leqslant \frac{1}{\eta}\sum_{i=1}^M u_i \log\frac{1}{\pi_i}\,;$$

inequality (11) is actually an improvement on this bound, which replaces the terms $\log\frac{1}{\pi_i}$ by $\log\frac{u_i}{\pi_i}$. Following Koolen et al. (2012), this refinement is used in Section 6.1 to obtain a tighter regret bound.

Appendix B. Proof of Theorem 3

Theorem 3 is in fact a corollary of the more general Proposition 10, valid in the specialist setting.

Proposition 10 *Assume we are given a set $\mathscr{M}$ of specialists, as well as a positive weight function $\pi : \mathscr{M} \to \mathbf{R}_+^*$. Assume that, at each time step $t \geqslant 1$, the set A_t of active specialists is finite. Then,*

denoting $A_{\leqslant t} = \bigcup_{1 \leqslant s \leqslant t} A_s$*, the aggregation of specialists*[4]

$$x_t = \frac{\sum_{i \in A_t} \pi(i)\, e^{-\eta\, L_{i,t-1}}\, x_{i,t}}{\sum_{i \in A_t} \pi(i)\, e^{-\eta\, L_{i,t-1}}} \tag{36}$$

achieves the following regret bound: for each $T \geqslant 1$ *and* $i \in \mathscr{M}$*, we have*

$$\sum_{t \leqslant T\,:\, i \in A_t} (\ell_t - \ell_{i,t}) \leqslant \frac{1}{\eta} \log\left(\frac{1}{\pi(i)} \sum_{j \in A_{\leqslant T}} \pi(j) \right). \tag{37}$$

Proof of Proposition 10 Fix $T \geqslant 1$, and denote $\Pi_T := \sum_{i \in A_{\leqslant T}} \pi(i)$. For $t = 1, \dots, T$, the forecast (36) may be rewritten as

$$x_t = \frac{\sum_{i \in A_t} \frac{\pi(i)}{\Pi_T}\, e^{-\eta\, L_{i,t-1}}\, x_{i,t}}{\sum_{i \in A_t} \frac{\pi(i)}{\Pi_T}\, e^{-\eta\, L_{i,t-1}}}$$

which corresponds precisely to the aggregation of the set of specialists $A_{\leqslant T}$ with prior weights $\pi(i)/\Pi_T$ and active specialists $A_t \subset A_{\leqslant T}$ (up to time T). (37) now follows from Proposition 2. ■

Proof of Theorem 3 It suffices to notice that the weights of GrowingHedge are, for $i \leqslant M_t$, $w_{i,t} = \pi_i\, e^{-\eta L_{i,t-1}}$ with $L_{i,t-1} = L_{\tau_i - 1} + \sum_{s=\tau_i}^{t} \ell_{i,s}$; hence, the forecasts of GrowingHedge are those of equation (36), and we can apply Proposition 10. ■

Appendix C. Proof of Lemma 4 and instantiations of algorithm MarkovHedge

Proof of Lemma 4 Denote, for each $t \geqslant 1$, $\pi^t(i_1, \dots, i_t) = \theta_1(i_1)\, \theta_2(i_2 \,|\, i_1) \cdots \theta_t(i_t \,|\, i_{t-1})$. Let $T \geqslant 1$ be arbitrary. We need to show that the predictions x_t of the exponentially weighted aggregation of sequences of experts i^T under the prior π^T at times $t = 1, \dots, T$ coincide with those of algorithm MarkovHedge.

First note that, by definition and since $L_{t-1}(i^T) = \sum_{s=1}^{t-1} \ell_{i_s,s} =: L_{t-1}(i^{t-1})$ does not depend on $i_t^T = (i_t, \dots, i_T)$, we have for $1 \leqslant t \leqslant T$

$$x_t = \frac{\sum_{i^T} \pi^T(i^T)\, e^{-\eta\, L_{t-1}(i^T)}\, x_t(i^T)}{\sum_{i^T} \pi^T(i^T)\, e^{-\eta\, L_{t-1}(i^T)}} = \frac{\sum_{i^t, i_{t+1}^T} \pi^T(i^t, i_{t+1}^T)\, e^{-\eta\, L_{t-1}(i^{t-1})}\, x_{i_t,t}}{\sum_{i^t, i_{t+1}^T} \pi^T(i^t, i_{t+1}^T)\, e^{-\eta\, L_{t-1}(i^{t-1})}}$$

$$\underset{(\star)}{=} \frac{\sum_{i^t} \pi^t(i^t)\, e^{-\eta\, L_{t-1}(i^{t-1})}\, x_{i_t,t}}{\sum_{i^t} \pi^t(i^t)\, e^{-\eta\, L_{t-1}(i^{t-1})}} = \frac{\sum_{i^{t-1}, i} \pi^t(i^{t-1}, i)\, e^{-\eta\, L_{t-1}(i^{t-1})}\, x_{i,t}}{\sum_{i^{t-1}, i} \pi^t(i^{t-1}, i)\, e^{-\eta\, L_{t-1}(i^{t-1})}}$$

where $(\star)$ is a consequence of the identity $\sum_{i_{t+1}^T} \pi^T(i^t, i_{t+1}^T) = \pi^t(i^t)$. Hence, denoting $w_t(i^t) := \pi^t(i^t)\, e^{-\eta L_{t-1}(i^{t-1})}$, we have

$$x_t = \frac{\sum_i \sum_{i^{t-1}} w_t(i^{t-1}, i)\, x_{i,t}}{\sum_i \sum_{i^{t-1}} w_t(i^{t-1}, i)} = \frac{\sum_{i=1}^M w_{i,t}\, x_{i,t}}{\sum_{i=1}^M w_{i,t}} = \sum_{i=1}^M v_{i,t}\, x_{i,t}$$

where we set $w_{i,t} := \sum_{i^{t-1}} \pi^t(i^{t-1}, i)\, e^{-\eta L_{t-1}(i^{t-1})}$ and $v_{i,t} := w_{i,t} / \big(\sum_{j=1}^M w_{j,t}\big)$. To conclude the proof, it remains to show that the weights $\boldsymbol{v}_t$ are those computed by algorithm MarkovHedge.

4. Denoting, as in equation (14), $L_{i,t} = \sum_{s \leqslant t\,:\, i \in A_s} \ell_{i,s} + \sum_{s \leqslant t\,:\, i \notin A_s} \ell_s$ for each specialist i and $t \geqslant 1$.

We proceed by induction on $t \geqslant 1$. For $t = 1$, we have for every $i = 1, \dots, M$, $w_{i,1} = w_1(i) = \pi^1(i) = \theta_1(i)$ and hence $v_{i,1} = \theta_1(i)$, *i.e.* $\boldsymbol{v}_1 = \boldsymbol{\theta}_1$. Moreover, for every $t \geqslant 1$, the identity $\pi^{t+1}(i^{t+1}) = \pi^t(i^t)\,\theta_{t+1}(i_{t+1} \,|\, i_t)$ implies

$$\begin{aligned} w_{t+1}(i^{t+1}) &= \pi^{t+1}(i^{t+1})\, e^{-\eta L_t(i^t)} \\ &= \theta_{t+1}(i_{t+1} \,|\, i_t)\, \pi^t(i^t)\, e^{-\eta L_{t-1}(i^{t-1})}\, e^{-\eta\, \ell_{i_t,t}} \\ &= \theta_{t+1}(i_{t+1} \,|\, i_t)\, w_t(i^t)\, e^{-\eta\, \ell_{i_t,t}} \end{aligned}$$

i.e., for every i, j and i^{t-1}, $w_{t+1}(i^{t-1}, j, i) = \theta_{t+1}(i \,|\, j)\, w_t(i^{t-1}, j)\, e^{-\eta\, \ell_{j,t}}$. Summing over i^{t-1} and j, this yields:

$$w_{i,t+1} = \sum_{j=1}^{M} \theta_{t+1}(i \,|\, j)\, w_{j,t}\, e^{-\eta\, \ell_{j,t}}\,. \tag{38}$$

Summing (38) over $i = 1, \dots, M$ gives $\sum_{i=1}^M w_{i,t+1} = \sum_{j=1}^M w_{j,t}\, e^{-\eta\, \ell_{j,t}}$ (since $\sum_{i=1}^M \theta_{t+1}(i \,|\, j) = 1$) and therefore

$$v_{i,t+1} = \frac{w_{i,t+1}}{\sum_{j=1}^M w_{j,t+1}} = \frac{\sum_{j=1}^M \theta_{t+1}(i \,|\, j)\, w_{j,t}\, e^{-\eta\, \ell_{j,t}}}{\sum_{j=1}^M w_{j,t}\, e^{-\eta\, \ell_{j,t}}} = \frac{\sum_{j=1}^M \theta_{t+1}(i \,|\, j)\, v_{j,t}\, e^{-\eta\, \ell_{j,t}}}{\sum_{j=1}^M v_{j,t}\, e^{-\eta\, \ell_{j,t}}} = \sum_{j=1}^{M} \theta_{t+1}(i \,|\, j)\, v_{j,t}^m$$

where $\boldsymbol{v}_t^m$ is the posterior distribution, defined by equation (9). This corresponds precisely to the update of the MarkovHedge algorithm, which completes the proof. ∎

We now instantiate the generic algorithm MarkovHedge and Proposition 5 on specific choices of prior weights and transition probabilities. This enables to recover a number of results from the literature. For concreteness, we take $\boldsymbol{\theta}_1 = \frac{1}{M}\mathbf{1}$.

Corollary 11 (Fixed share) *Setting $\theta_t(i \,|\, j) = (1-\alpha)\,\mathbf{1}_{i=j} + \alpha\,\frac{1}{M}$ with $\alpha \in (0,1)$, this leads to the Fixed-Share algorithm of Herbster and Warmuth (1998) with update $\boldsymbol{v}_{t+1} = (1-\alpha)\,\boldsymbol{v}_t^m + \alpha\,\frac{1}{M}\mathbf{1}$ and regret bound*

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_{i_t,t} \leqslant \frac{k+1}{\eta} \log M + \frac{k}{\eta} \log \frac{1}{\alpha} + \frac{T-k-1}{\eta} \log \frac{1}{1-\alpha}\,, \tag{39}$$

where $k = k(i^T)$ denotes the number of shifts, $1 < \sigma_1 < \cdots < \sigma_k \leqslant T$ these shifts (such that $i_{\sigma_j} \neq i_{\sigma_j - 1}$) and $\sigma_0 = 1$. When T and k are fixed and known, this bound is minimized by choosing $\alpha = \frac{k}{T-1}$ and becomes, denoting $H(p) = -p \log p - (1-p)\log(1-p)$ the binary entropy function,

$$\frac{k+1}{\eta} \log M + \frac{T-1}{\eta} H\Big(\frac{k}{T-1}\Big) \leqslant \frac{k+1}{\eta} \log M + \frac{k}{\eta} \log \frac{T-1}{k} + \frac{k}{\eta}\,. \tag{40}$$

Remark 9 *The quantity of equation (40), i.e. the bound on the regret of fully tuned Fixed Share algorithm, is essentially equal to the optimal bound $\frac{1}{\eta} \log \binom{T-1}{k} M^{k+1} \approx \frac{k+1}{\eta} \log M + \frac{k}{\eta} \log \frac{T-1}{k}$, obtained by aggregating all sequences of experts with at most k shifts (which would require to maintain a prohibitively large number of weights).*

Corollary 12 (Decreasing share) *Consider the special case of algorithm MarkovHedge where $\theta_t(i \,|\, j) = (1-\alpha_t)\,\mathbf{1}_{i=j} + \frac{\alpha_t}{M}$, so that the update becomes $\boldsymbol{v}_{t+1} = (1-\alpha_{t+1})\,\boldsymbol{v}_t^m + \frac{\alpha_{t+1}}{M}\,\mathbf{1}$. For*

every $T \geqslant 1$, $0 \leqslant k \leqslant T$, and every sequence of experts $i^T = (i_1, \dots, i_T)$ with k shifts at times $\sigma_1 < \cdots < \sigma_k$,

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_{i_t,t} \leqslant \frac{k+1}{\eta} \log M + \frac{1}{\eta} \sum_{j=1}^{k} \log \frac{1}{\alpha_{\sigma_j}} + \frac{1}{\eta} \sum_{t=2}^{T} \log \frac{1}{1-\alpha_t} \tag{41}$$

In the special case[5] *when $\alpha_t = \frac{1}{t}$, this bound becomes, for* every *T, k and i^T:*

$$\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_{i_t,t} \leqslant \frac{k+1}{\eta} \log M + \frac{1}{\eta} \sum_{j=1}^{k} \log \sigma_j + \frac{1}{\eta} \log T \leqslant \frac{k+1}{\eta} \log M + \frac{k+1}{\eta} \log T\,. \tag{42}$$

Remark 10 *The result of Corollary 12 is worth emphasizing: at no computational overhead, the use of decreasing transition probabilities gives a bound essentially in $\frac{1}{\eta}(k+1)\log M + \frac{1}{\eta} k \log T$ valid for every T and k, which is close to the bound $\frac{1}{\eta}(k+1)\log M + \frac{1}{\eta} k \log \frac{T}{k}$ one gets by optimally tuning α as a function of T and k in the Fixed Share algorithm, particularly when $k \ll T$ (in this latter case of rare shifts, the first, sharper bound of equation (42) is even more appealing).*

Proof of corollaries 11 and 12 We consider the Decreasing Share algorithm, with time-varying transition probabilities $\alpha_t \in (0,1)$ (the Fixed Share algorithm corresponds to the special case $\alpha_t = \alpha$). Let $i^T = (i_1, \dots, i_T)$ be a sequence of experts with shifts at times $\sigma_1 < \cdots < \sigma_k$. By Proposition 5, we have

$$\begin{aligned}
\sum_{t=1}^{T} \ell_t - \sum_{t=1}^{T} \ell_{i_t,t} &\leqslant \frac{1}{\eta} \log \frac{1}{1/M} + \frac{1}{\eta} \sum_{j=1}^{k} \log \frac{1}{\alpha_{\sigma_j}/M} + \frac{1}{\eta} \sum_{t \neq \sigma_j} \log \frac{1}{1-\alpha_{\sigma_j}+\alpha_{\sigma_j}/M} \\
&\leqslant \frac{k+1}{\eta} \log M + \frac{1}{\eta} \sum_{j=1}^{k} \log \frac{1}{\alpha_{\sigma_j}} + \frac{1}{\eta} \sum_{t \neq \sigma_j} \log \frac{1}{1-\alpha_{\sigma_j}}
\end{aligned}$$

Corollary 11 directly follows by taking $\alpha_t = \alpha$ in the above inequality, whereas the bound (41) of Corollary 12 is obtained by bounding $\sum_{t \neq \sigma_j} \log \frac{1}{1-\alpha_t} \leqslant \sum_{t=2}^{T} \log \frac{1}{1-\alpha_t}$. In the case when $\alpha_t = \frac{1}{t}$, we recover (42) by substituting for α_t and noting that

$$\sum_{t=2}^{T} \log \frac{1}{1-1/t} = \sum_{t=2}^{T} \log \frac{t}{t-1} = \log T\,. \tag{43}$$

■

Appendix D. Proof of Proposition 8

Proof Since $x_t(i,1) = x_{i,t}$ and $x_t(i,0) = x_t$, equation (31) implies that the forecast x_t of SleepingMarkovHedge satisfies:

$$x_t = \sum_{i=1}^{M} \sum_{a \in \{0,1\}} v_t(i,a)\, x_t(i,a)\,.$$

Hence, SleepingMarkovHedge reduces to algorithm MarkovHedge over the sleeping experts, *i.e.* (by Lemma 4, up to time T) to the exponentially weighted aggregation of sequences of sleeping

5. Which we consider because of the simplicity of the bound as well as its proof, involving a telescoping simplification; it is akin to Theorem 10 of Koolen and de Rooij (2013), which uses $\alpha_t = 1 - e^{-c/t}$.

experts under the Markov prior $\pi((i,a_t)_{1\leqslant t\leqslant T}) = \theta_{i,1}(a_1)\prod_{t=2}^{T}\theta_{i,t}(a_t\,|\,a_{t-1})$ (and 0 for other sequences). Hence, if u is the uniform probability on the n sequences $(e_p,a_{p,t})_{1\leqslant t\leqslant T}$, $1\leqslant p\leqslant n$, we have by Proposition 1:

$$\sum_{\iota^T} u(\iota)\,(L_T - L_T(\iota^T)) \leqslant \frac{1}{\eta}\Delta(u\,\|\,\pi) = \frac{1}{\eta}\frac{1}{n}\sum_{p=1}^{n}\log\frac{1/n}{\pi((e_p,a_{p,t})_{1\leqslant t\leqslant T})} \tag{44}$$

As shown in the reformulation of the regret with respect to sparse sequences of experts of Section 6.1, the left hand side of equation (44) equals $\frac{1}{n}(L_T - L_T(i^T))$. The desired regret bound (34) follows by substituting for π in the right-hand side. ∎

Appendix E. Uniform bounds and optimality

In this section, we provide simple bounds derived from Theorems 3, 6, 7 and 9 that are not quite as adaptive to the parameters of the comparison class as the ones provided in Section 2, but are more uniform and hence more interpretable. We then discuss the optimality of these bounds, by relating them either to theoretical lower bounds or to information-theoretic upper bounds (obtained by naively aggregating all elements of the comparison class, which is computationally prohibitive).

Constant experts Consider the algorithm GrowingHedge with the uniform (unnormalized) prior: $\pi_i = 1$ for each $i\geqslant 1$. By Theorem 3, this algorithm achieves the regret bound

$$\frac{1}{\eta}\log M_T$$

with respect to each constant expert.

This regret bound cannot be improved in general: indeed, consider the logarithmic loss on $\mathbf{N}^*$, defined by $\ell(x,y) = -\log x(y)$ for every $y\in\mathbf{N}^*$ and every probability distribution x on $\mathbf{N}^*$. Fix $T\geqslant 1$, and consider the sequence $y_t = x_{i,t} = 1$ ($1\leqslant t<T$, $1\leqslant i\leqslant M_t$) and $y_t\in\{1,\dots,M_T\}$ and $x_{i,T} = i$ for $i = 1,\dots,M_T$. For each $i = 1,\dots,M_T$, we have $\sup_{1\leqslant i\leqslant M_T}(L_T - L_{i,T}) = \sup_{1\leqslant i\leqslant M_T} -\log\frac{x_t(y_t)}{x_{i,t}(y_t)} = -\log x_t(y_t)$. Now whatever x_t is, there exists $y_t\in\{1,\dots,M_T\}$ such that $x_t(y_t)\leqslant\frac{1}{M_T}$ (since x_t sums to 1). Since y_t is picked by an adversary after x_t is chosen, the adversary can always ensure a regret of at least $\log M_T$.

Fresh sequences of experts By Theorems 3 and 6, algorithms GrowingHedge and FreshMarkovHedge with a uniform prior ($\pi_i = 1$ for each $i\geqslant 1$) achieve the regret bound

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta}\sum_{j=1}^{k}\log M_{\sigma_j-1} + \frac{1}{\eta}\log M_T$$

for every sequence i^T of fresh experts with shifts at times $\boldsymbol{\sigma} = (\sigma_1,\dots,\sigma_k)$. By the same argument as above, this bound cannot be improved in general.

Arbitrary admissible sequences of experts By Theorem 7, algorithm GrowingMarkovHedge with uniform prior $\boldsymbol{\pi}$ and transition probabilities $\alpha_t = \frac{1}{t}$ achieves, for every admissible sequence i^T

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta}\sum_{j=0}^{k}\log M_{\sigma_{j+1}-1} + \frac{1}{\eta}\sum_{j=1}^{k_1}\log\sigma_j^1 + \frac{1}{\eta}\log T \leqslant \frac{1}{\eta}(k+1)\log M_T + \frac{1}{\eta}(k_1+1)\log T\,.$$

where $\boldsymbol{\sigma}^0 = (\sigma_1^0, \dots, \sigma_{k_0}^0)$ (resp. $\boldsymbol{\sigma}^1 = (\sigma_1^1, \dots, \sigma_{k_1}^1)$) denotes the shifts to fresh (resp. incumbent) experts, with $k = k_0 + k_1$.

This simple bound is close to the information-theoretic bound obtained by aggregating all admissible sequences of experts: indeed, the number of such sequences is bounded by (with equality if $M_T = M_1$) $M_T^{k+1}\binom{T-1}{k_1}$ (an admissible sequence is determined by its switches to fresh experts – at most $M_T^{k_0+1}$ possibilities – and its switches to incumbent experts – at most $M_T^{k_1}$ possibilities for the choices of the experts, and at most $\binom{T-1}{k_1}$ choices for the switches to incumbent experts). The regret bound corresponding to the aggregation of this large expert class is therefore of order

$$\frac{1}{\eta} \log M_T^{k+1} \binom{T-1}{k_1} \approx \frac{1}{\eta}(k+1) \log M_T + \frac{1}{\eta} k_1 \log \frac{T-1}{k_1} \,,$$

which is close to the bound of GrowingHedge, especially if $k_1 \ll T$.

Sparse admissible sequences Finally, Theorem 9 implies that algorithm GrowingSleepingMarkovHedge, with uniform weights $\boldsymbol{\pi}$ and transition probabilities $\alpha_t = \beta_t = \frac{1}{t \log t}$, has a regret bound of

$$L_T - L_T(i^T) \leqslant \frac{1}{\eta} n \log \frac{M_T}{n} + \frac{1}{\eta} n (\log 2 + c_T \log\log T) + \frac{2}{\eta} k \log T + \frac{1}{\eta} 2k \log\log T \,.$$

for any sparse admissible sequence i^T with at most k shifts and taking values in a pool of n experts, where $c_T := (\log\log T)^{-1} \sum_{t=2}^T \log \frac{1}{1-\alpha_t} \longrightarrow_{T\to\infty} 1$. Again, for $k \ll T$, this is close to the information-theoretic upper bound obtained by aggregating all sparse sequences with k shifts in a pool of n experts, of approximately $n \log \frac{M_T}{n} + (k+1) \log n + k \log \frac{T}{k}$. The main difference, namely the doubling of the term $k \log T$ in the regret bound of GrowingSleepingMarkovHedge, is not specific to the growing experts setting, and also appears in the context of a fixed set of experts (Bousquet and Warmuth, 2002; Koolen et al., 2012).