Exploiting Easy Data in Online Optimization

Amir Sani Gergely Neu Alessandro Lazaric
{amir.sani,gergely.neu,alessandro.lazaric}@inria.fr
SequeL team, INRIA Lille – Nord Europe, France

Near-optimal guarantees on both "**Easy**" and "**Hard**" data!

What is "Easy" Data?

Examples:
- Highly predictable sequences
- IID losses with large gaps between means
- Strongly convex losses
- See Settings 2, 3 and 4 in Figure 1.

Algorithms:
- Variants of Follow-the-leader (FTL)
- Typical regret guarantees: $\mathcal{O}(\log T)$

Problems:
- Horrible performance on 'Hard' data!
- See Setting 1 in Figure 1.

What is "Hard" Data?

Examples:
- Non-IID Adversarial Losses
- Non-Stationary distributions
- Small gaps between means
- Non-strongly convex losses
- See Setting 1 in Figure 1.

Algorithms:
- Variants of Follow-the-regularized-leader (FTRL)
- Typical regret guarantees: $\mathcal{O}(\sqrt{T})$

Problems:
- Horrible performance on 'Easy' data.
- See Settings 2 and 3 in Figure 1.

Online Optimization (OO) [3]

Parameters:
- Decision set $\mathcal{S}$
- Number of rounds T
- Family of loss functions $\mathcal{F} \subseteq \mathcal{S}^{[0,1]}$

For all $t = 1, 2, \ldots, T$, **repeat**
1. Environment chooses loss function $f_t \in \mathcal{F}$.
2. Learner chooses a decision $x_t \in \mathcal{S}$.
3. Environment reveals f_t.
4. Learner suffers loss $f_t(x_t)$.

Competing against a Benchmark

Our method guarantees a constant regret w.r.t. any existing **benchmark** strategy together with small regret against the **best strategy** in hindsight. This is particularly useful in domains where the learning algorithm should be **safe** and **never worsen** the performance of an existing strategy (e.g., portfolio optimization with benchmark reference).

$(\mathcal{AB})$-Prod

Parameters:
- Learning rate $\eta \in (0, 1/2]$
- Initial weights $w_{1,\mathcal{A}} = \eta$ and $w_{1,\mathcal{B}} = 1 - \eta$
- Rounds T

For all $t = 1, 2, \ldots, T$, **repeat**
1. Let
$$s_t = \frac{w_{t,\mathcal{A}}}{w_{t,\mathcal{A}} + w_{1,\mathcal{B}}}.$$
2. Observe a_t from Algorithm $\mathcal{A}$
3. Observe b_t from Benchmark $\mathcal{B}$
4. Predict
$$x_t = \begin{cases} a_t & \text{with probability } s_t, \\ b_t & \text{otherwise.} \end{cases}$$
5. Observe f_t and suffer loss $f_t(x_t)$.
6. Feed f_t to $\mathcal{A}$ and $\mathcal{B}$.
7. Compute $\delta_t = f_t(b_t) - f_t(a_t)$ and set
$$w_{t+1,\mathcal{A}} = w_{t,\mathcal{A}} \cdot (1 + \eta\delta_t).$$

Anytime $(\mathcal{AB})$-Prod

Parameters:
- Learning rate $\eta_1 = 1/2$
- Initial weights $w_{1,\mathcal{A}} = w_{1,\mathcal{B}} = 1/2$
- Rounds T

For all $t = 1, 2, \ldots, T$, **repeat**
1. Let
$$\eta_t = \sqrt{\frac{1}{1 + \sum_{s=1}^{t-1} (f_s(b_s) - f_s(a_s))^2}}$$
and
$$s_t = \frac{\eta_t w_{t,\mathcal{A}}}{\eta_t w_{t,\mathcal{A}} + w_{1,\mathcal{B}}/2}.$$
2. Observe a_t from Algorithm $\mathcal{A}$
3. Observe b_t from Benchmark $\mathcal{B}$
4. Predict
$$x_t = \begin{cases} a_t & \text{with probability } s_t, \\ b_t & \text{with probability } 1 - s_t. \end{cases}$$
5. Observe f_t and suffer loss $f_t(x_t)$.
6. Feed f_t to $\mathcal{A}$ and $\mathcal{B}$.
7. Compute $\delta_t = f_t(b_t) - f_t(a_t)$ and set
$$w_{t+1,\mathcal{A}} = w_{t,\mathcal{A}} \cdot (1 + \eta_{t-1}\delta_t)^{\eta_t/\eta_{t-1}}.$$

Theoretical Results

Theorem 1 (cf. Lemma 1 in [2])

For any assignment of the loss sequence, the total expected loss of $(\mathcal{AB})$-Prod initialized with weights $w_{1,\mathcal{B}} \in (0,1)$ and $w_{1,\mathcal{B}} = 1 - w_{1,\mathcal{A}}$ simultaneously satisfies
$$\widehat{L}_T\big((\mathcal{AB})\text{-Prod}\big) \le \widehat{L}_T(\mathcal{A}) + \eta \sum_{t=1}^{T} \big(f_t(b_t) - f_t(a_t)\big)^2 - \frac{\log w_{1,\mathcal{A}}}{\eta}$$
and
$$\widehat{L}_T\big((\mathcal{AB})\text{-Prod}\big) \le \widehat{L}_T(\mathcal{B}) - \frac{\log w_{1,\mathcal{B}}}{\eta}.$$

Corollary 1

Let $C \ge 1$ be an upper bound on the total benchmark loss $\widehat{L}_T(\mathcal{B})$. Then setting $\eta = 1/2 \cdot \sqrt{(\log C)/C} < 1/2$ and $w_{1,\mathcal{B}} = 1 - w_{1,\mathcal{A}} = 1 - \eta$ simultaneously guarantees
$$\mathfrak{R}_T\big((\mathcal{AB})\text{-Prod}\big) \le \mathfrak{R}_T(\mathcal{A}) + 2\sqrt{C \log C}$$
for any $x \in \mathcal{S}$ and
$$\mathfrak{R}_T\big((\mathcal{AB})\text{-Prod}\big) \le \mathfrak{R}_T(\mathcal{A}) + 2 \log 2$$
against any assignment of the loss sequence.

Setting	$\mathcal{S}$	$\mathcal{F}$	A	B	"Hard" Regret	"Easy" Regret	"Easy" Data
Prediction with Expert Advice	Δ_N	$[0,1]^N$	FTL	Hedge	$\mathcal{O}\left(\sqrt{T \log(NT)}\right)$	$\mathcal{O}(\log T)$	IID
Tracking the Best Expert[a]	Δ_N	$[0,1]^N$	FTL(w)	FixedShare	$\mathcal{O}\left(\sqrt{KT \log(NT)}\right)$	$\mathcal{O}(K \log T)$	Piecewise IID
Online Convex Optimization	Convex Closed Subset of $\mathbb{R}^d$	SCF[b]	FTL	OGD	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\log T)$[c]	Strongly Convex
Two-Point Bandit	$\{1, 2, \ldots, T\}$	$[0,1]^N$	EXP3	UCB	$\mathcal{O}\left(\sqrt{NT \log(NT)}\right)$	$\mathcal{O}(\log T)$	IID

Lemma 1

Assume a partition of $[1, T]$ into K intervals exists such that the losses are generated i.i.d. within each interval. Furthermore, assume the expectation of losses on the best expert within each interval is at least δ away from the expected loss of all other experts. Then, setting $w = \lceil 4 \log(NT/K)/\delta^2 \rceil$, the regret of FTL($w$) is upper bounded for any $y_{1:T}$ as
$$\mathbb{E}\Big[\mathfrak{R}_T(\text{FTL}(w), y_{1:T})\Big] \le \frac{4K}{\delta^2} \log(NT/K) + 2K,$$
where the expectation is taken with respect to the distribution of the losses.

[a] Solves the open problem of learning on "Easy" and "Hard" loss sequences in the tracking the best expert setting proposed by [4].
[b] smooth convex functions
[c] Matching the performance of AOGD [5]

$(\mathcal{AB})$-Prod Proof

Let
- $W_t = w_{t,\mathcal{A}} + w_{t,\mathcal{B}}$,
- $\ell_{t,\mathcal{A}} = f_t(a_t)$, $\ell_{t,\mathcal{B}} = f_t(b_t)$,
- $\hat{\ell}_{t,i} = \ell_{t,i} - \ell_{t,\mathcal{B}}$ for $i \in \{\mathcal{A}, \mathcal{B}\}$.

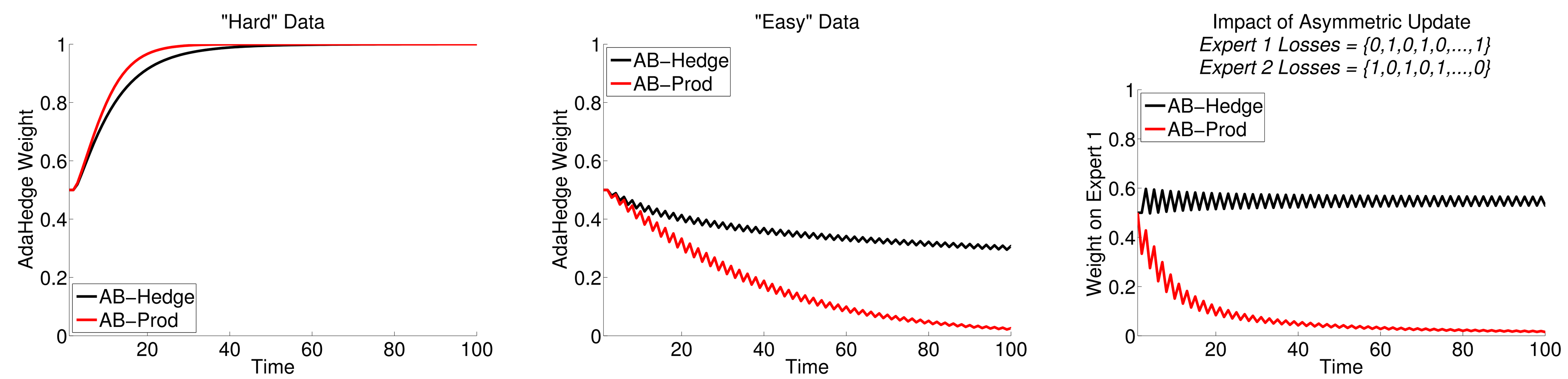
For $i \in \{\mathcal{A}, \mathcal{B}\}$,
$$\log \frac{W_{T+1}}{W_1} \ge \log w_{T+1,i} = \log w_{1,i} + \sum_{t=1}^{T} \log(1 - \eta\hat{\ell}_{t,i})$$
$$\ge \log w_{1,i} - \eta \sum_{t=1}^{T} \hat{\ell}_{t,i} - \eta^2 \sum_{t=1}^{T} \hat{\ell}_{t,i}^2,$$
where we used that $\log(1 - x) \ge -x - x^2$ holds for all $x \le \frac{1}{2}$. Furthermore, for any $t = 1, 2, \ldots, T$ we have
$$\log \frac{W_{t+1}}{W_t} = \log\left(\sum_i \frac{w_{t,i}(1 - \eta\hat{\ell}_{t,i})}{W_t}\right)$$
$$= \log\left(1 - \eta \sum_i p_{t,i}\hat{\ell}_{t,i}\right) \le -\eta \sum_i p_{t,i}\hat{\ell}_{t,i},$$
by $\log(1 - x) \le -x$. Summing up for all t, combining the above inequalities and using the definition of $\hat{\ell}_{t,i}$, we get for $i = \mathcal{A}$ that
$$\widehat{L}_T\big((\mathcal{AB})\text{-Prod}\big) - \widehat{L}_T(\mathcal{A}) \le \eta \sum_{t=1}^{T} (\ell_{t,\mathcal{A}} - \ell_{t,\mathcal{B}})^2 - \frac{\log w_{1,\mathcal{A}}}{\eta}.$$
Similarly, for $i = \mathcal{B}$, we obtain
$$\widehat{L}_T\big((\mathcal{AB})\text{-Prod}\big) - \widehat{L}_T(\mathcal{B}) \le -\frac{\log w_{1,\mathcal{B}}}{\eta}.$$

$(\mathcal{AB})$-Hedge Proof

Let $w_{t+1,\mathcal{A}} = w_{t,\mathcal{A}} \cdot e^{\eta\delta_t}$ and
- $W_t = w_{t,\mathcal{A}} + w_{t,\mathcal{B}}$,
- $\ell_{t,\mathcal{A}} = f_t(a_t)$, $\ell_{t,\mathcal{B}} = f_t(b_t)$,
- $\hat{\ell}_{t,i} = \ell_{t,i} - \ell_{t,\mathcal{B}}$ for $i \in \{\mathcal{A}, \mathcal{B}\}$.

For $i \in \{\mathcal{A}, \mathcal{B}\}$,
$$\log \frac{W_{T+1}}{W_1} \ge \log w_{T+1,i}$$
$$= \log w_{1,i} - \eta \sum_{t=1}^{T} \hat{\ell}_{t,i},$$
where we used the definition of the update rule. Furthermore, for any $t = 1, 2, \ldots, T$ we have
$$\log \frac{W_{t+1}}{W_t} = \log\left(\sum_i \frac{w_{t,i} e^{-\eta\hat{\ell}_{t,i}}}{W_t}\right)$$
$$\le \log\left(1 - \eta \sum_i p_{t,i}\hat{\ell}_{t,i} + \frac{\eta^2}{8}\right) \le -\eta \sum_i p_{t,i}\hat{\ell}_{t,i} + \frac{\eta^2}{8},$$
by Hoeffding's lemma. Summing up for all t, combining the above inequalities and using the definition of $\hat{\ell}_{t,i}$, we get for $i = \{\mathcal{A}, \mathcal{B}\}$ that
$$\widehat{L}_T\big((\mathcal{AB})\text{-Hedge}\big) - \widehat{L}_T(i) \le \frac{\eta T}{8} - \frac{\log w_{1,i}}{\eta}.$$

Secret Sauce



Empirical Results

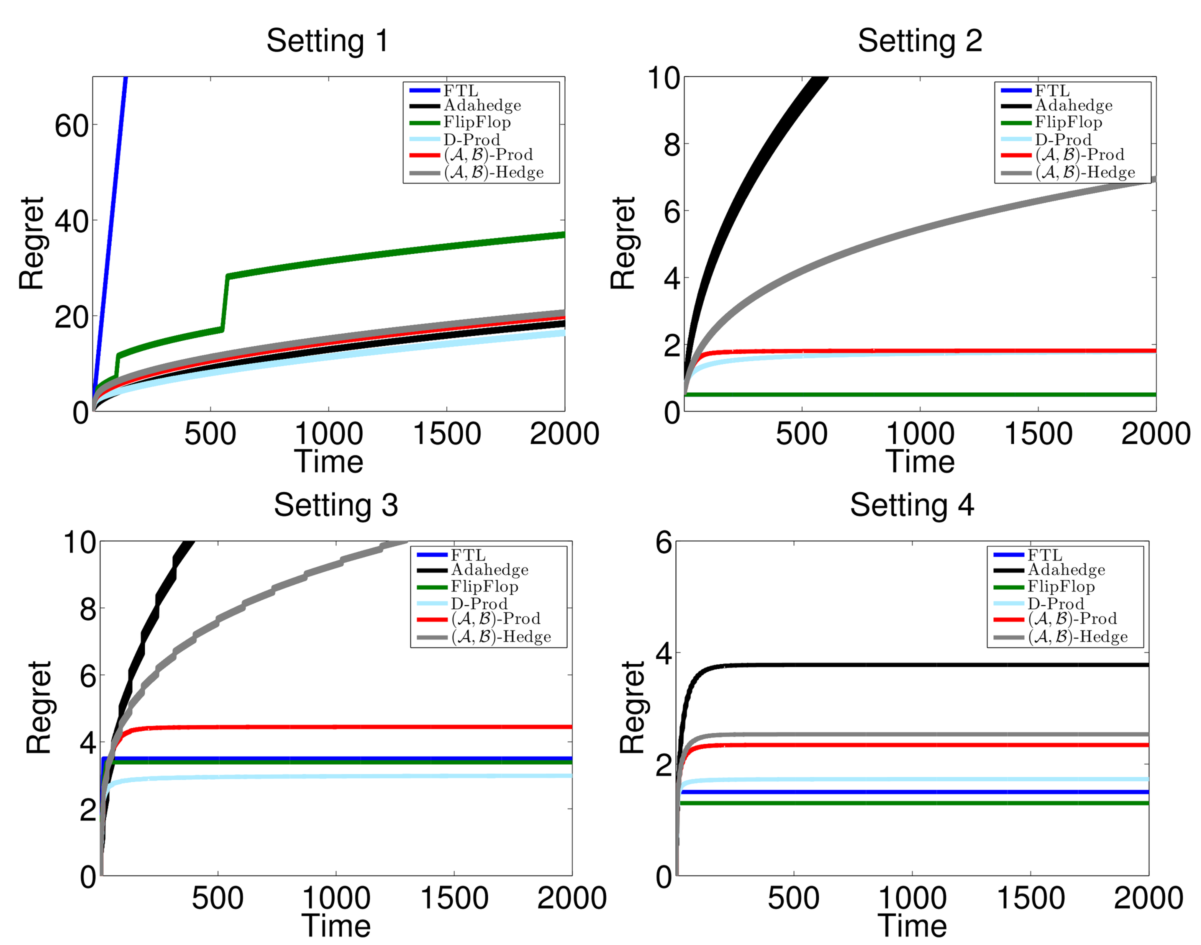


Figure : Hand tuned loss sequences from [1]

Discussion

- **Summary**
  - Given a learning algorithm $\mathcal{A}$, with worst-case performance guarantees, and an opportunistic strategy $\mathcal{B}$, exploiting a specific structure within the loss sequence, smoothly adapts to "Easy" and "Hard" problems.
  - Guarantees best performance between benchmark $\mathcal{B}$ and a worst-case algorithm $\mathcal{A}$
  - General-purpose, Interpretable, Simple
- **Open Problems**
  - Learning with temporal constraints (e.g., switching costs, MDPs)?
  - What are good benchmark strategies for easy data?
  - Learning with partial feedback?

References

[1] de Rooij,S.,Van Erven,T.,Grunwald, P.D.,& Koolen,W.M. (2013).Follow the leader if you can, hedge if you must.

[2] Even-Dar, E., Kearns, M., Mansour, Y., & Wortman, J. (2008). Regret to the best vs. regret to the average. Machine Learning, 72(1-2), 21-37.

[3] Zinkevich, M. (2003). Online convex programming and generalized infinitesimal gradient ascent.

[4] Koolen, W. M., & Warmuth, M. K. Shifting Experts on Easy Data.

[5] Bartlett, P. L., Hazan, E., & Rakhlin, A. (2008). Adaptive online gradient descent.