Learning the Learning Rate for Prediction with Expert Advice

Wouter M. Koolen Tim van Erven Peter D. Grünwald

QUT Queensland University of Technology

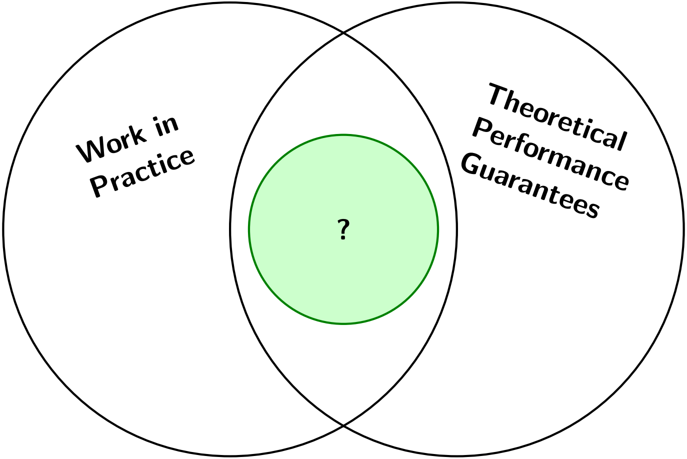
Berkeley UNIVERSITY OF CALIFORNIA

CWI Centrum Wiskunde & Informatica

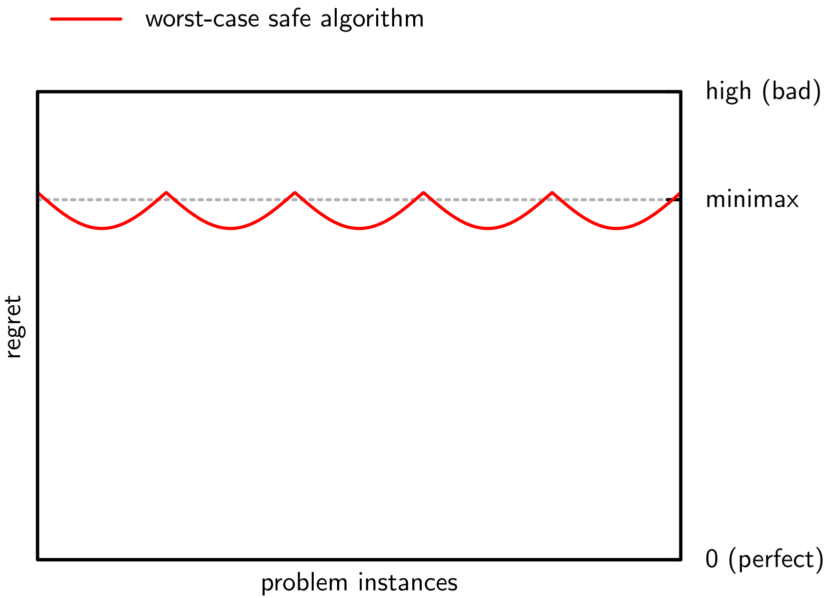
Universiteit Leiden

Lorentz Workshop Leiden, Thursday 20th November, 2014

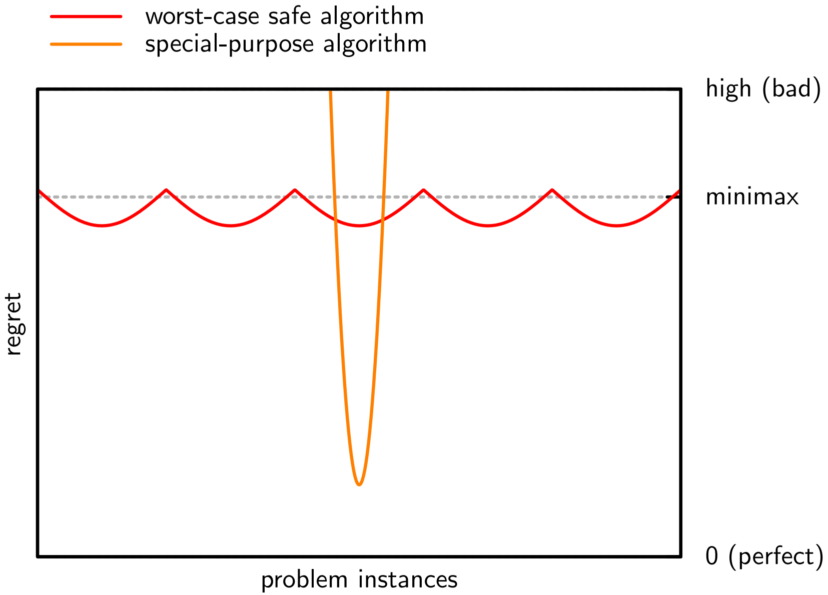
Online Learning Algorithms



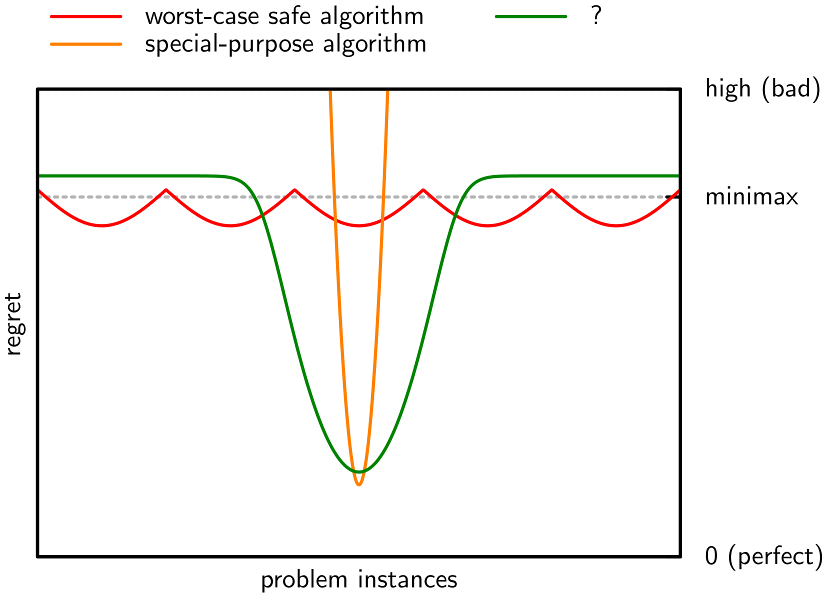
Learning as a Game



Practice is not Adversarial



Luckiness



worst-case safe algorithm
special-purpose algorithm
?
high (bad)
minimax
0 (perfect)
regret
problem instances

Fundamental model for learning: Hedge setting

- K experts

...

Fundamental model for learning: Hedge setting

- K experts

$\dots$

- In round $t = 1, 2, \dots$
  - Learner plays distribution $\boldsymbol{w}_t = (w_t^1, \dots, w_t^K)$ on experts
  - Adversary reveals expert losses $\boldsymbol{\ell}_t = (\ell_t^1, \dots, \ell_t^K) \in [0, 1]^K$

  - Learner incurs loss $\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t$

Fundamental model for learning: Hedge setting

- K experts

$\dots$

- In round $t = 1, 2, \dots$
  - Learner plays distribution $\boldsymbol{w}_t = (w_t^1, \dots, w_t^K)$ on experts
  - Adversary reveals expert losses $\boldsymbol{\ell}_t = (\ell_t^1, \dots, \ell_t^K) \in [0,1]^K$

  - Learner incurs loss $\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t$

- Evaluation criterion is the **regret**:

$$\mathcal{R}_T \;:=\; \underbrace{\sum_{t=1}^{T} \boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t}_{\text{Learner}} - \underbrace{\min_k \sum_{t=1}^{T} \ell_t^k}_{\text{best expert}}$$

Canonical algorithm for the Hedge setting

Hedge algorithm **with learning rate** η:

$$w_t^k := \frac{e^{-\eta L_{t-1}^k}}{\sum_k e^{-\eta L_{t-1}^k}} \qquad \text{where} \qquad L_{t-1}^k = \sum_{s=1}^{t-1} \ell_s^k.$$

Canonical algorithm for the Hedge setting

Hedge algorithm **with learning rate η**:

$$w_t^k := \frac{e^{-\eta L_{t-1}^k}}{\sum_k e^{-\eta L_{t-1}^k}} \qquad \text{where} \qquad L_{t-1}^k = \sum_{s=1}^{t-1} \ell_s^k.$$

The tuning $\eta = \eta^{\text{worst case}} := \sqrt{\frac{8 \ln K}{T}}$ results in

$$\mathcal{R}_T \leq \sqrt{T/2 \ln K}$$

and we have matching lower bounds.

Canonical algorithm for the Hedge setting

Hedge algorithm **with learning rate** η:

$$w_t^k \;:=\; \frac{e^{-\eta L_{t-1}^k}}{\sum_k e^{-\eta L_{t-1}^k}} \qquad \text{where} \qquad L_{t-1}^k \;=\; \sum_{s=1}^{t-1} \ell_s^k.$$

The tuning $\eta = \eta^{\text{worst case}} := \sqrt{\frac{8 \ln K}{T}}$ results in

$$\mathcal{R}_T \;\leq\; \sqrt{T/2 \ln K}$$

and we have matching lower bounds.

Case closed?

Gap between Theory and Practice

Practitioners report that tuning $\eta \gg \eta^{\text{worst case}}$ works much better. [DGGS13]

Theory and Practice getting closer

Practitioners report that tuning $\eta \gg \eta^{\text{worst case}}$ works much better. [DGGS13]

Series of worst-case **data-dependent** improvements

$$\mathcal{R}_T \;\leq\; \sqrt{T/2 \ln K}$$

$$\boxed{T} \longrightarrow \boxed{L_T^*} \longrightarrow \boxed{V_T} \longrightarrow \boxed{\Delta_T}$$

Theory and Practice getting closer

Practitioners report that tuning $\eta \gg \eta^{\text{worst case}}$ works much better. [DGGS13]

Series of worst-case **data-dependent** improvements

$$\mathcal{R}_T \;\leq\; \sqrt{T/2 \ln K} \qquad \boxed{T} \longrightarrow \boxed{L_T^*} \longrightarrow \boxed{V_T} \longrightarrow \boxed{\Delta_T}$$

and **extension** to scenarios where Follow-the-Leader ($\eta = \infty$) shines (IID losses)

$$\mathcal{R}_T \;\leq\; \min\left\{\mathcal{R}_T^{\text{worst case}}, \mathcal{R}_T^{\infty}\right\}$$

Theory and Practice getting closer

Practitioners report that tuning $\eta \gg \eta^{\text{worst case}}$ works much better. [DGGS13]

Series of worst-case **data-dependent** improvements

$$\mathcal{R}_T \;\leq\; \sqrt{T/2\ln K} \qquad \boxed{T} \longrightarrow \boxed{L_T^*} \longrightarrow \boxed{V_T} \longrightarrow \boxed{\Delta_T}$$

and **extension** to scenarios where Follow-the-Leader ($\eta = \infty$) shines (IID losses)

$$\mathcal{R}_T \;\leq\; \min\left\{\mathcal{R}_T^{\text{worst case}}, \mathcal{R}_T^{\infty}\right\}$$

Case closed?

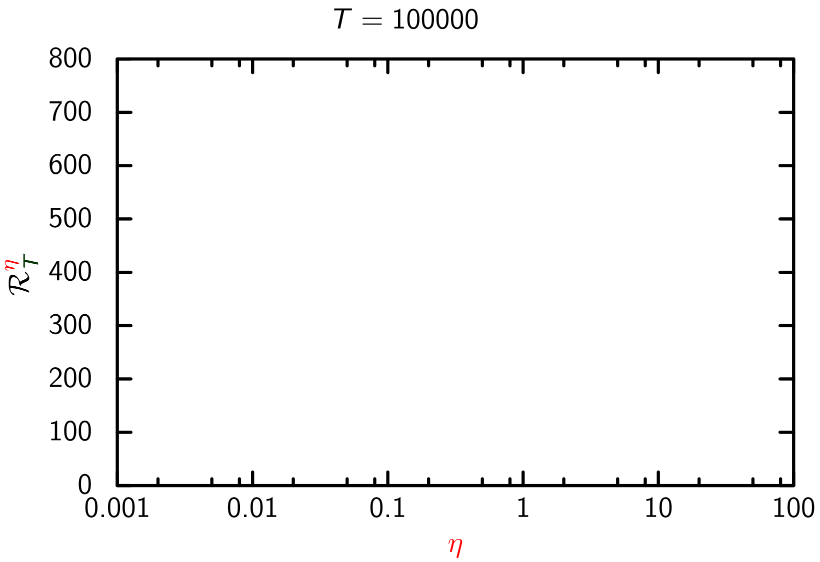
Menu

Grand goal: be almost as good as best learning rate η

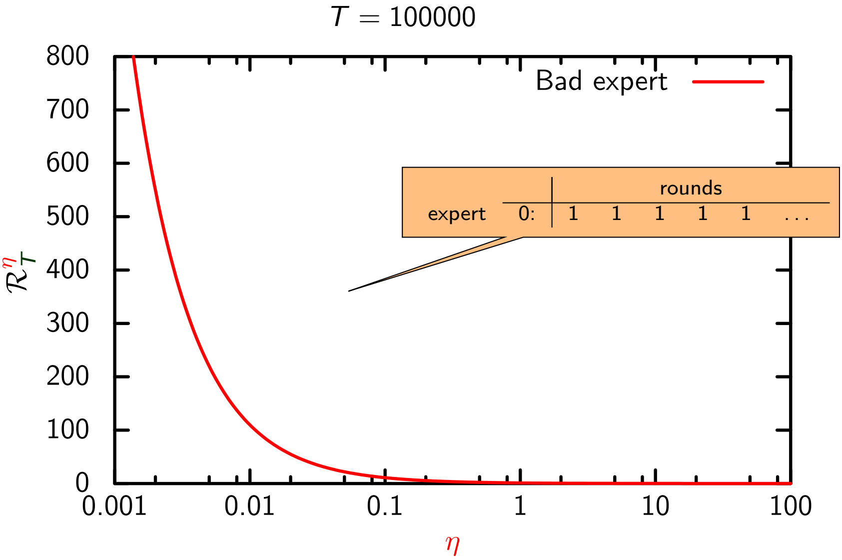
$$\mathcal{R}_T \;\approx\; \min_\eta \mathcal{R}_T^\eta.$$

- Example problematic data
- Key ideas

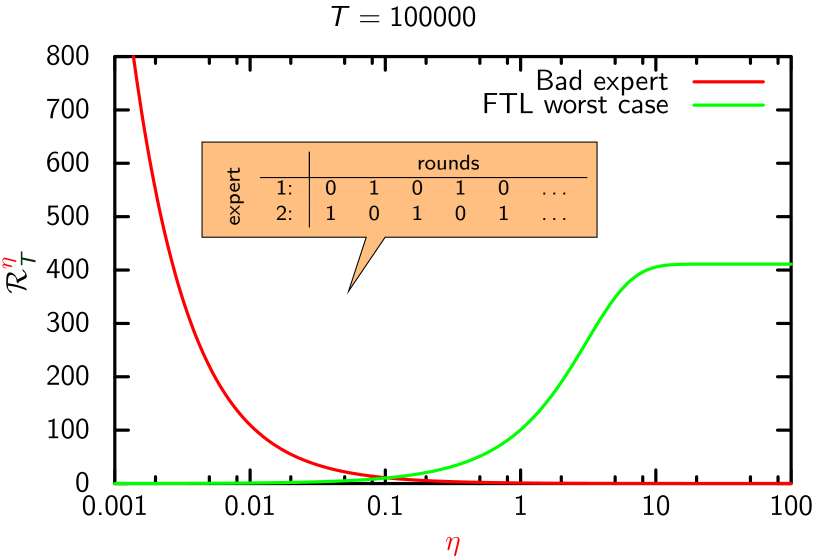
Current η tunings miss the boat



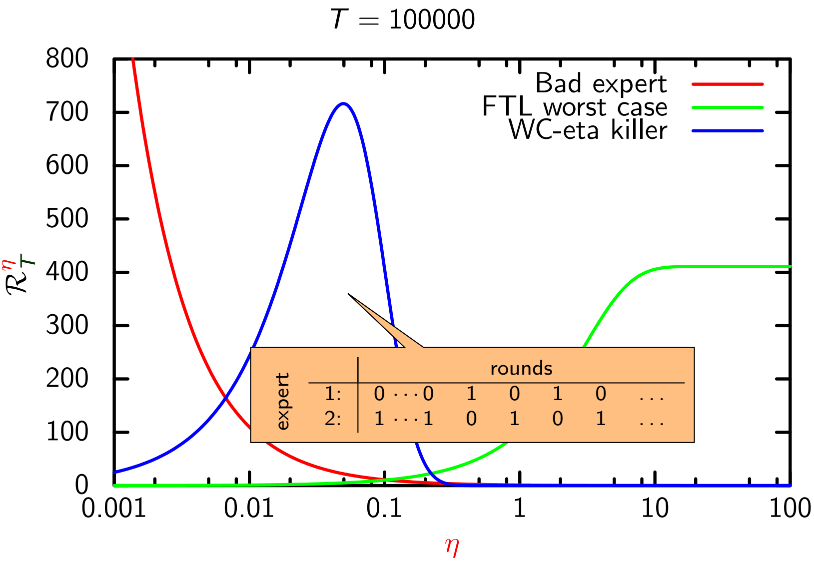
Current η tunings miss the boat



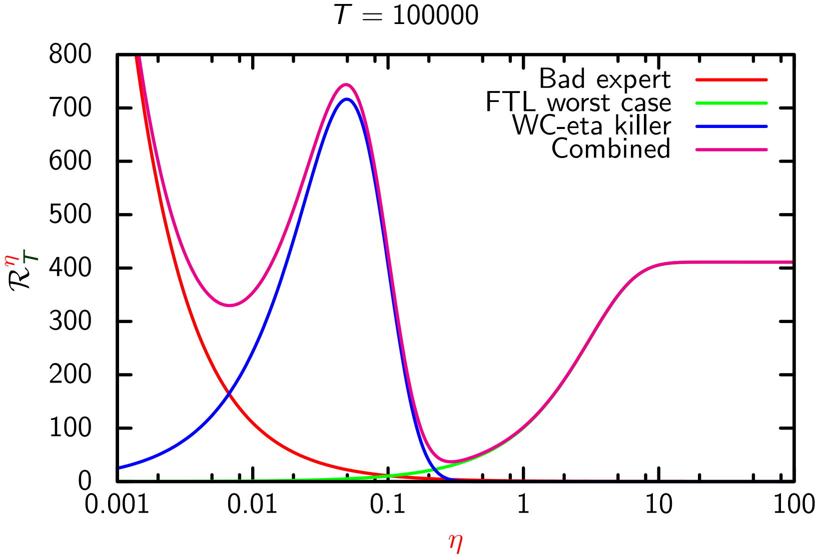
Current η tunings miss the boat



Current η tunings miss the boat



Current η tunings miss the boat



LLR algorithm in a nutshell

LLR

- maintains a **finite grid** $\eta^1, \ldots, \eta^{i_{\mathsf{max}}}, \eta^{\mathsf{ah}}$
- cycles over the grid. For each η^i:
  - Play the η^i **Hedge weights**
  - Evaluate η^i by its **mixability gap**
  - Until its **budget** doubled
- adds next lower grid point on demand

LLR algorithm in a nutshell

LLR

- maintains a **finite grid** $\eta^1, \dots, \eta^{i_{\max}}, \eta^{\text{ah}}$
- cycles over the grid. For each η^i:
  - Play the η^i **Hedge weights**
  - Evaluate η^i by its **mixability gap**
  - Until its **budget** doubled
- adds next lower grid point on demand

Resources:

- Time: $O(K)$ per round (same as Hedge).
- Memory: $O(\ln T) \to O(1)$.

Unavoidable notation

$$\begin{aligned} h_t &= \boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t, && \text{(Hedge loss)} \\ m_t &= \frac{-1}{\eta} \ln \sum_k w_t^k e^{-\eta \ell_t^k}, && \text{(Mix loss)} \\ \delta_t &= h_t - m_t. && \text{(Mixability gap)} \end{aligned}$$

Unavoidable notation

$$
\begin{aligned}
h_t &= \boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t, && \text{(Hedge loss)} \\
m_t &= \frac{-1}{\eta} \ln \sum_k w_t^k e^{-\eta \ell_t^k}, && \text{(Mix loss)} \\
\delta_t &= h_t - m_t. && \text{(Mixability gap)}
\end{aligned}
$$

And capitals denote cumulatives

$$
\Delta_T = \sum_{t=1}^{T} \delta_t, \ldots
$$

Key Idea 1: Monotone regret lower bound

Problem: Regret $\mathcal{R}_T^\eta$ is **not** increasing with T.

But we have a monotone lower bound:

$$\mathcal{R}_T^\eta \;\geq\; \Delta_T^\eta$$

Proof:

$$\mathcal{R}_t^\eta \;=\; H_T - L_T^* \;=\; \underbrace{H_T - M_T}_{\text{mixability gap}} + \underbrace{M_T - L_T^*}_{\text{mix loss regret}}$$

Now use

$$M_T \;=\; \frac{-1}{\eta} \ln\left(\sum_k \frac{1}{K} e^{-\eta L_T^k}\right) \in L_T^* + \left[0, \frac{\ln K}{\eta}\right]$$

Upshot: measure quality of each η using cumulative mixability gap.

Key Idea 2: Grid of η suffices

For $\gamma \geq 1$:

$$\delta_t^{\gamma\eta} \;\leq\; \gamma e^{(\gamma-1)(\ln K+\eta)} \delta_t^{\eta}$$

I.e. δ_t^{η} cannot be much better than $\delta_t^{\gamma\eta}$.

Exponentially spaced grid of η suffices.

Key Idea 3: Lowest η is "AdaHedge"

AdaHedge:

$$\eta_t^{\text{ah}} := \frac{\ln K}{\Delta_{t-1}^{\text{ah}}}$$

Result:

$$\mathcal{R}_T \leq \sum_{i=1}^{i_{\max}} \Delta_T^i + c\Delta_T^{\text{ah}}$$

Key Idea 4: Budgeted timesharing

Active grid points

$$\eta^1, \qquad \eta^2, \qquad \ldots, \qquad \eta^{i_{\max}}, \qquad \eta_t^{\mathrm{ah}}$$

with (heavy-tailed) prior distribution

$$\pi^1, \qquad \pi^2, \qquad \ldots, \qquad \pi^{i_{\max}}, \qquad \pi^{\mathrm{ah}}$$

LLR maintains **invariant**:

$$\frac{\Delta_T^1}{\pi^1} \approx \frac{\Delta_T^2}{\pi^2} \approx \ldots \approx \frac{\Delta_T^{i_{\max}}}{\pi^{i_{\max}}} \approx \frac{\Delta_T^{\mathrm{ah}}}{\pi^{\mathrm{ah}}}$$

Run each η_i in turn until its cumulative mixability gap $\frac{\Delta_T^i}{\pi^i}$ doubled.

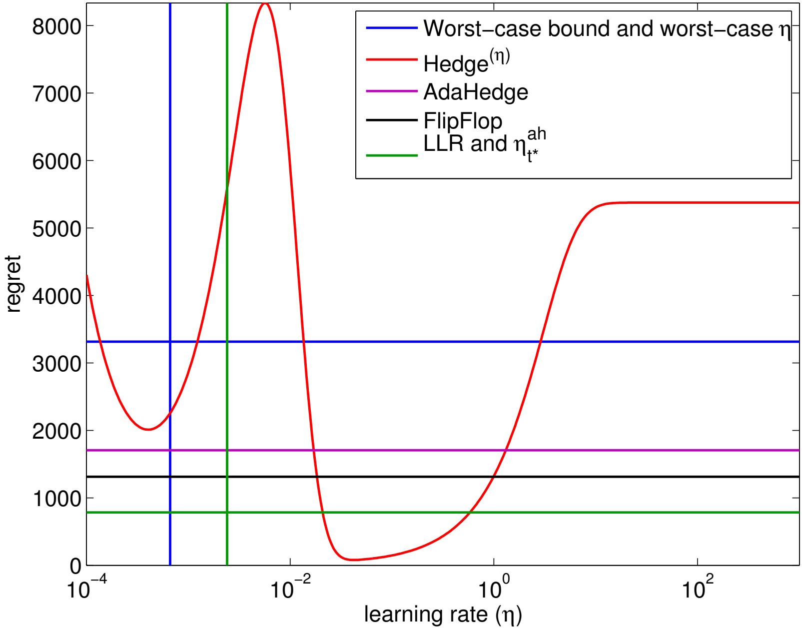
$$\sum_{i=1}^{i_{\max}} \Delta_T^i = \sum_{i=1}^{i_{\max}} \pi^i \frac{\Delta_T^i}{\pi^i} \approx \frac{\Delta_T^j}{\pi^j} \sum_{i=1}^{i_{\max}} \pi^i \leq \frac{\Delta_T^j}{\pi^j}$$

Putting it all together

Two bounds:

$$\mathcal{R}_T \;\leq\; \tilde{O} \begin{cases} \ln K \ln \frac{1}{\eta} \mathcal{R}_T^{\eta} & \text{for all } \eta \in [\eta_{t^*}^{\mathsf{ah}}, 1] \\ \\ \mathcal{R}_T^{\infty} \end{cases}$$

Run on synthetic data ($T = 2 \cdot 10^7$)



Conclusion

- Higher learning rates often achieve lower regret
  - In practice
  - Constructed data
- Learning the Learning Rate (LLR) algorithm
  - Performance close to best learning rate in hindsight

Conclusion

- Higher learning rates often achieve lower regret
  - In practice
  - Constructed data
- Learning the Learning Rate (LLR) algorithm
  - Performance close to best learning rate in hindsight

Open problems:

- LLR as PoC

  Can we do it simpler, prettier, smoother and tighter?

Thank you!

Marie Devaine, Pierre Gaillard, Yannig Goude, and Gilles Stoltz.
Forecasting electricity consumption by aggregating specialized experts; a review of the sequential aggregation of specialized experts, with an application to Slovakian and French country-wide one-day-ahead (half-)hourly predictions.
Machine Learning, 90(2):231–260, February 2013.