Learning Faster from Easy Data

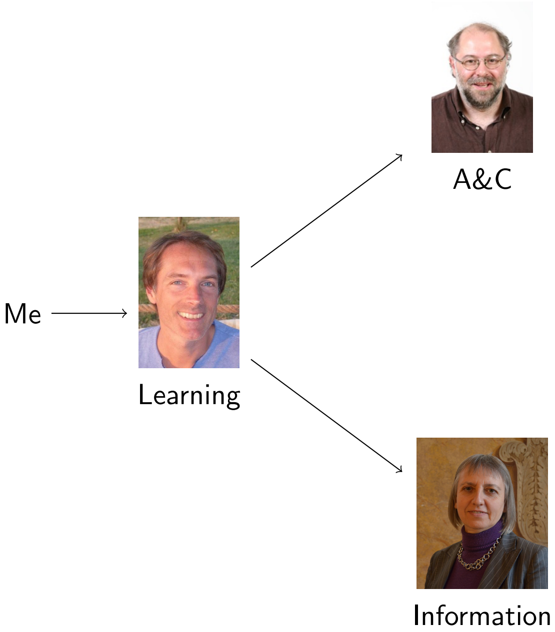
Wouter M. Koolen

CWI Scientific Meeting, Friday 27th November, 2015

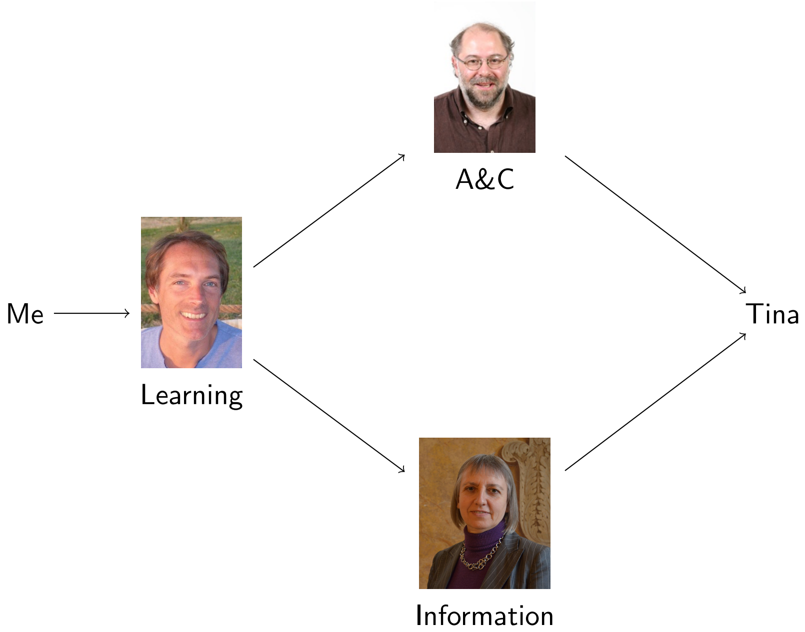
Bio

06–11	PhD	CWI Centrum Wiskunde & Informatica		
11–13	Postdoc	Royal Holloway University of London		
13–15	Postdoc	QUT Queensland University of Technology	and	Berkeley UNIVERSITY OF CALIFORNIA
15–	VENI	CWI Centrum Wiskunde & Informatica		

Organogram

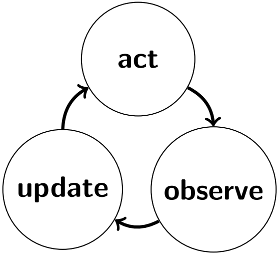


Organogram



This talk: online learning

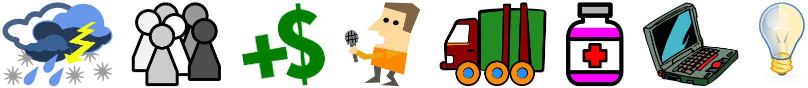
Sequential decision making protocol



Definition

To **learn** X = **act** as if you already know best $x \in X$

Typical online learning applications



- Invest like best stock (or portfolio)
- Predict demand like best linear regressor (Amazon)
- Commute like best route (OSP)
- Compress like best variable-order markov model (CTW)
- Tracking the best electricity consumption forecasting company (EDF)
- ...

Applications outside online learning comfort-zone

- Convex optimisation, both online, and batch (SGD).

- Computing Nash equilibria in two-player zero-sum games
- Game play (Monte Carlo Tree Search, e.g. for Go)
- Boosting
- Differential Privacy
- A/B testing
- Predictive complexity (algorithmic information theory)
- ...

Fundamental model for learning: Hedge setting

- K experts

. . .

Fundamental model for learning: Hedge setting

- K experts

...

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

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

Fundamental model for learning: Hedge setting

- K experts

…

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

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

- The goal is to have small **regret**

$$R_T^k \;:=\; \underbrace{\sum_{t=1}^T \boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t}_{\text{Learner}} - \underbrace{\sum_{t=1}^T \ell_t^k}_{\text{Expert } k}$$

with respect to every expert k.

Classic Hedge Result

The **Hedge** algorithm with **learning rate** η

$$w_{t+1}^k := \frac{e^{-\eta L_t^k}}{\sum_k e^{-\eta L_t^k}} \qquad \text{where} \qquad L_t^k = \sum_{s=1}^{t} \ell_s^k,$$

upon proper tuning of η ensures [Freund and Schapire, 1997]

$$R_T^k \prec \sqrt{T \ln K} \qquad \text{for each expert } k$$

which is tight for adversarial (worst-case) losses

Classic Hedge Result

The **Hedge** algorithm with **learning rate** η

$$w_{t+1}^k := \frac{e^{-\eta L_t^k}}{\sum_k e^{-\eta L_t^k}} \qquad \text{where} \qquad L_t^k = \sum_{s=1}^{t} \ell_s^k,$$

upon proper tuning of η ensures [Freund and Schapire, 1997]

$$R_T^k \prec \sqrt{T \ln K} \qquad \text{for each expert } k$$

which is tight for adversarial (worst-case) losses

but **underwhelming** in practice

Classic Hedge Result

The **Hedge** algorithm with **learning rate** η

$$w_{t+1}^k := \frac{e^{-\eta L_t^k}}{\sum_k e^{-\eta L_t^k}} \qquad \text{where} \qquad L_t^k = \sum_{s=1}^t \ell_s^k,$$

upon proper tuning of η ensures [Freund and Schapire, 1997]

$$R_T^k \prec \sqrt{T \ln K} \qquad \text{for each expert } k$$

which is tight for adversarial (worst-case) losses

but **underwhelming** in practice

- Why?
- Practitioners report good performance with ad-hoc η danger
- Can we do better?

Beyond the Worst Case

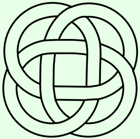
Two reasons data is often **easier** in practice:

Beyond the Worst Case

Two reasons data is often **easier** in practice:

Data complexity

- Stochastic data (gap)
- Low noise
- Low variance

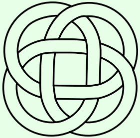


Beyond the Worst Case

Two reasons data is often **easier** in practice:

Data complexity

- Stochastic data (gap)
- Low noise
- Low variance



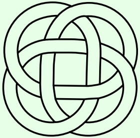
second-order

Beyond the Worst Case

Two reasons data is often **easier** in practice:

Data complexity

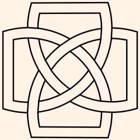
- Stochastic data (gap)
- Low noise
- Low variance



second-order

Model complexity

- Simple model is good
- Multiple good models

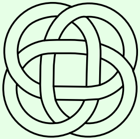


Beyond the Worst Case

Two reasons data is often **easier** in practice:

Data complexity

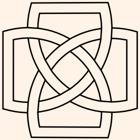
- Stochastic data (gap)
- Low noise
- Low variance



second-order

Model complexity

- Simple model is good
- Multiple good models



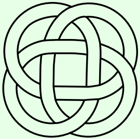
quantiles

Beyond the Worst Case

Two reasons data is often **easier** in practice:

Data complexity

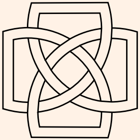
- Stochastic data (gap)
- Low noise
- Low variance



second-order

Model complexity

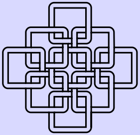
- Simple model is good
- Multiple good models



quantiles

Second-order & Quantiles

- Any combination

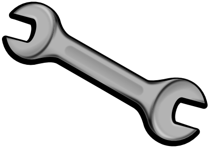


All we need is the right learning rate

Existing algorithms

(Hedge, Prod, ...)

with

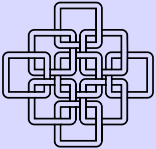


oracle

learning rate η

exploit

Sec-ord. & Quant.

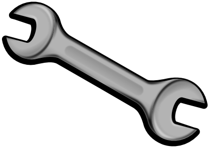


All we need is the right learning rate

Existing algorithms

(Hedge, Prod, . . .)

with

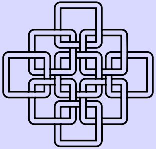


oracle

learning rate η

exploit

Sec-ord. & Quant.



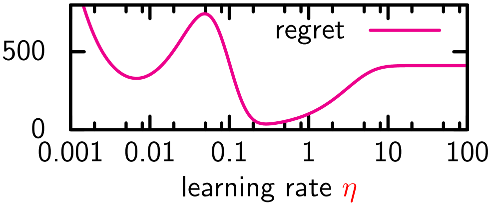
Can we exploit Second-order & Quantiles **on-line**?

But everyone struggles with the learning rate

Oracle η

- **not** monotonic,
- **not** smooth

over time.

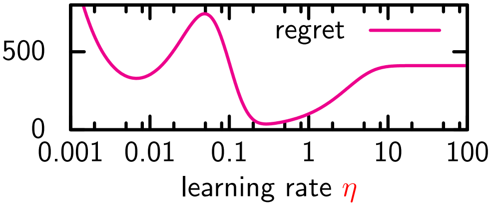


But everyone struggles with the learning rate

Oracle η

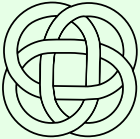
- **not** monotonic,
- **not** smooth

over time.



State of the art:

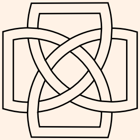
Second-order



Cesa-Bianchi, Mansour, and Stoltz 2007, Hazan and Kale 2010, Chiang, Yang, Lee, Mahdavi, Lu, Jin, and Zhu 2012, De Rooij, Van Erven, Grünwald, and Koolen 2014, Gaillard, Stoltz, and Van Erven 2014, Steinhardt and Liang 2014

or

Quantiles



Hutter and Poland 2005, Chaudhuri, Freund, and Hsu 2009, Chernov and Vovk 2010, Luo and Schapire 2014

Learning the learning rate

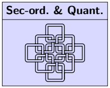
With Tim van Erven: New framework for algorithm design where simply **putting a prior** γ on η and integrating it out works.

Our algorithm **Squint**

$$w_{t+1}^{k} \;\propto\; \pi(k) \mathop{\mathbb{E}}_{\gamma(\eta)} \left[e^{\eta R_t^k - \eta^2 V_t^k} \eta \right]$$

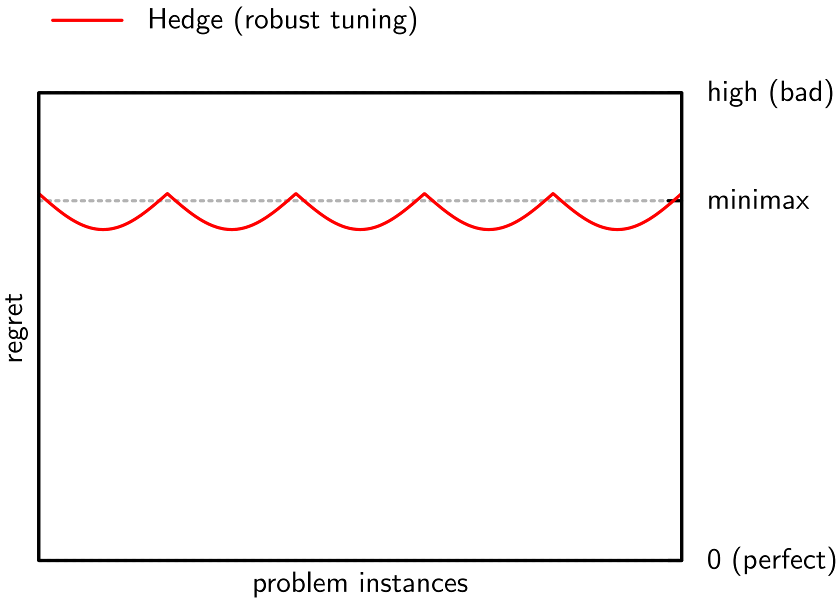
guarantees for each subset $\mathcal{K}$ of experts, at each time $T \geq 0$:

$$R_T^{\mathcal{K}} \;\prec\; \sqrt{V_T^{\mathcal{K}} \left(-\ln \pi(\mathcal{K})\right)}$$

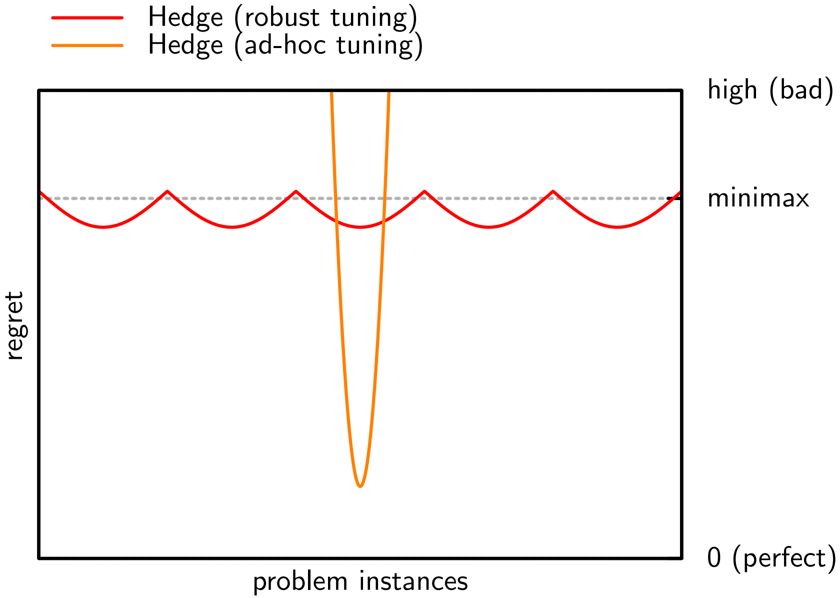


- Run-time of Hedge

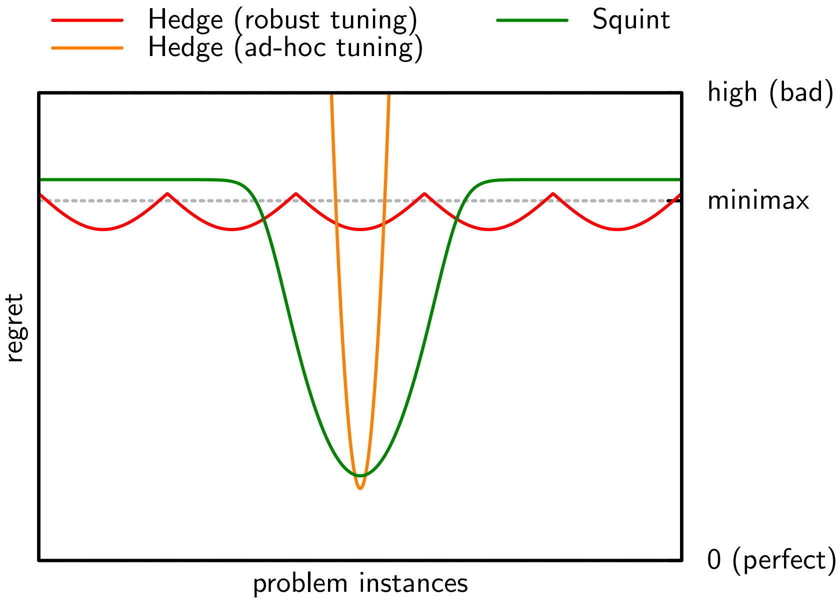
Summary


Hedge (robust tuning)
high (bad)
minimax
0 (perfect)
regret
problem instances

Summary


Hedge (robust tuning)
Hedge (ad-hoc tuning)
high (bad)
minimax
0 (perfect)
regret
problem instances

Summary


Hedge (robust tuning)
Hedge (ad-hoc tuning)
Squint
high (bad)
minimax
0 (perfect)
regret
problem instances

Conclusion

Fresh algorithm for fundamental learning task

- new "different" perspective
- same efficiency
- adaptive (better) guarantees

Currently scaling up to advanced learning tasks

- Combinatorial games
- Matrix games
- Online optimization (gradient descent)

- Very welcome to discuss further
- Try it out

`http://bitbucket.org/wmkoolen/squint`

Thank you!