

MetaGrad: Multiple Learning Rates in Online Learning

Tim van Erven



Universiteit
Leiden

Joint work with: Wouter Koolen, Peter Grünwald

UvA, May 16, 2017

Online Learning Example: Electricity Forecasting



- ▶ Every day t an electricity company needs to predict how much electricity Y_t is needed the next day
- ▶ Given feature vector $\mathbf{X}_t \in \mathbb{R}^d$, predict $\hat{Y}_t = \langle \mathbf{w}_t, \mathbf{X}_t \rangle$ with a linear model
- ▶ Next day: observe Y_t
- ▶ Measure loss by $\ell_t(\mathbf{w}_t) = (Y_t - \hat{Y}_t)^2$ and improve parameter estimates: $\mathbf{w}_t \rightarrow \mathbf{w}_{t+1}$

Online Convex Optimization

Parameters \mathbf{w} take values in a convex domain \mathcal{U}

- 1: **for** $t = 1, 2, \dots, T$ **do**
- 2: Learner estimates $\mathbf{w}_t \in \mathcal{U}$
- 3: Environment reveals convex loss function $\ell_t : \mathcal{U} \rightarrow \mathbb{R}$
- 4: Learner incurs loss $\ell_t(\mathbf{w}_t)$, observes gradient $\mathbf{g}_t = \nabla \ell_t(\mathbf{w}_t)$
- 5: **end for**

Online Convex Optimization

Parameters \mathbf{w} take values in a convex domain \mathcal{U}

- 1: **for** $t = 1, 2, \dots, T$ **do**
- 2: Learner estimates $\mathbf{w}_t \in \mathcal{U}$
- 3: Environment reveals convex loss function $\ell_t : \mathcal{U} \rightarrow \mathbb{R}$
- 4: Learner incurs loss $\ell_t(\mathbf{w}_t)$, observes gradient $\mathbf{g}_t = \nabla \ell_t(\mathbf{w}_t)$
- 5: **end for**

Example: Classification with Convex Surrogate Losses

Given $\mathbf{X}_t \in \mathbb{R}^d$, predict label $Y_t \in \{-1, +1\}$

$$\ell_t(\mathbf{w}) = \max\{0, 1 - Y_t \langle \mathbf{w}, \mathbf{X}_t \rangle\} \quad (\text{hinge loss})$$

$$\ell_t(\mathbf{w}) = \ln \left(1 + e^{-Y_t \langle \mathbf{w}, \mathbf{X}_t \rangle} \right) \quad (\text{logistic loss})$$

$$\ell_t(\mathbf{w}) = (Y_t - \langle \mathbf{w}, \mathbf{X}_t \rangle)^2 \quad (\text{squared loss})$$

Online Convex Optimization

Parameters w take values in a convex domain \mathcal{U}

- 1: **for** $t = 1, 2, \dots, T$ **do**
- 2: Learner estimates $w_t \in \mathcal{U}$
- 3: Environment reveals convex loss function $\ell_t : \mathcal{U} \rightarrow \mathbb{R}$
- 4: Learner incurs loss $\ell_t(w_t)$, observes gradient $g_t = \nabla \ell_t(w_t)$
- 5: **end for**

Minimize **regret** w.r.t. oracle parameters $u \in \mathcal{U}$:

$$\text{Regret}_T^u = \sum_{t=1}^T \ell_t(w_t) - \sum_{t=1}^T \ell_t(u)$$

Assumptions: $\text{diameter}(\mathcal{U}) \leq 1$, $\|g_t\|_2 \leq 1$.

Standard Methods

Online Gradient Descent (OGD)

Move into direction of steepest descent:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{g}_t \quad (\text{and project onto } \mathcal{U} \text{ if go outside})$$

Step size determined by **learning rate** $\eta_t > 0$.

Standard Methods

Online Gradient Descent (OGD)

Move into direction of steepest descent:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{g}_t \quad (\text{and project onto } \mathcal{U} \text{ if go outside})$$

Step size determined by learning rate $\eta_t > 0$.

Online Newton Step (ONS)

Make less sensitive to parametrization by running OGD on pre-conditioned functions $\ell_t(\Sigma_{t+1}^{1/2} \tilde{\mathbf{w}})$:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \Sigma_{t+1} \mathbf{g}_t,$$

where $\Sigma_{t+1} = (I + 2\eta^2 \sum_{s=1}^t \mathbf{g}_s \mathbf{g}_s^\top)^{-1}$ and η fixed.

The Standard Picture

Minimax rates based on curvature [Hazan, 2016]:

Convex ℓ_t	\sqrt{T}	OGD with $\eta_t \propto \frac{1}{\sqrt{t}}$
Strongly convex ℓ_t	$\ln T$	OGD with $\eta_t \propto \frac{1}{t}$
Exp-concave ℓ_t	$d \ln T$	ONS with $\eta \propto 1$

- ▶ **Strongly convex:** second derivative at least $\alpha > 0$, implies exp-concave
- ▶ **Exp-concave:** $e^{-\alpha \ell_t}$ concave
Satisfied by logistic loss, squared loss, but not hinge loss

The Standard Picture

Minimax rates based on curvature [Hazan, 2016]:

Convex ℓ_t	\sqrt{T}	OGD with $\eta_t \propto \frac{1}{\sqrt{t}}$
Strongly convex ℓ_t	$\ln T$	OGD with $\eta_t \propto \frac{1}{t}$
Exp-concave ℓ_t	$d \ln T$	ONS with $\eta \propto 1$

Limitations:

- ▶ Different method in each case. (Requires sophisticated users.)
- ▶ Theoretical tuning of η_t **very conservative**
- ▶ What if curvature varies between rounds?
- ▶ In many applications data are **stochastic** (i.i.d.) Should be easier than worst case...

Need Adaptive Methods!

Main Idea

Existing Adaptivity Results:

- ▶ [Bartlett et al., 2007], [Do et al., 2009]
Adaptive GD: **strongly convex + general convex**
- ▶ Other types of adaptivity: [Orabona, 2014, Orabona and Pál, 2016, Duchi et al., 2011, Hazan and Kale, 2010, Orabona et al., 2015]
- ▶ In all cases, tuning of learning rate η is crucial!

Main Idea

Existing Adaptivity Results:

- ▶ [Bartlett et al., 2007], [Do et al., 2009]
Adaptive GD: **strongly convex + general convex**
- ▶ Other types of adaptivity: [Orabona, 2014, Orabona and Pál, 2016, Duchi et al., 2011, Hazan and Kale, 2010, Orabona et al., 2015]
- ▶ In all cases, tuning of learning rate η is crucial!

Key idea:

- ▶ So let's learn optimal η from the data!

Main obstacle:

- ▶ Avoid learning η at slow rate itself.

Breakthrough:

- ▶ **Multiple Eta Gradient Algorithm (MetaGrad)**
- ▶ First method to aggregate multiple learning rates

MetaGrad Algorithm

η_1



η_2



η_3



η_4

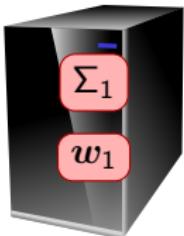


$$\dots \underbrace{\frac{1}{2} \ln(T)}_{\leq 16}$$

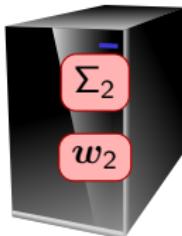


MetaGrad Algorithm

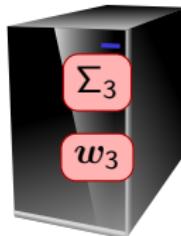
η_1



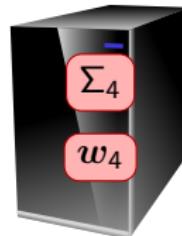
η_2



η_3



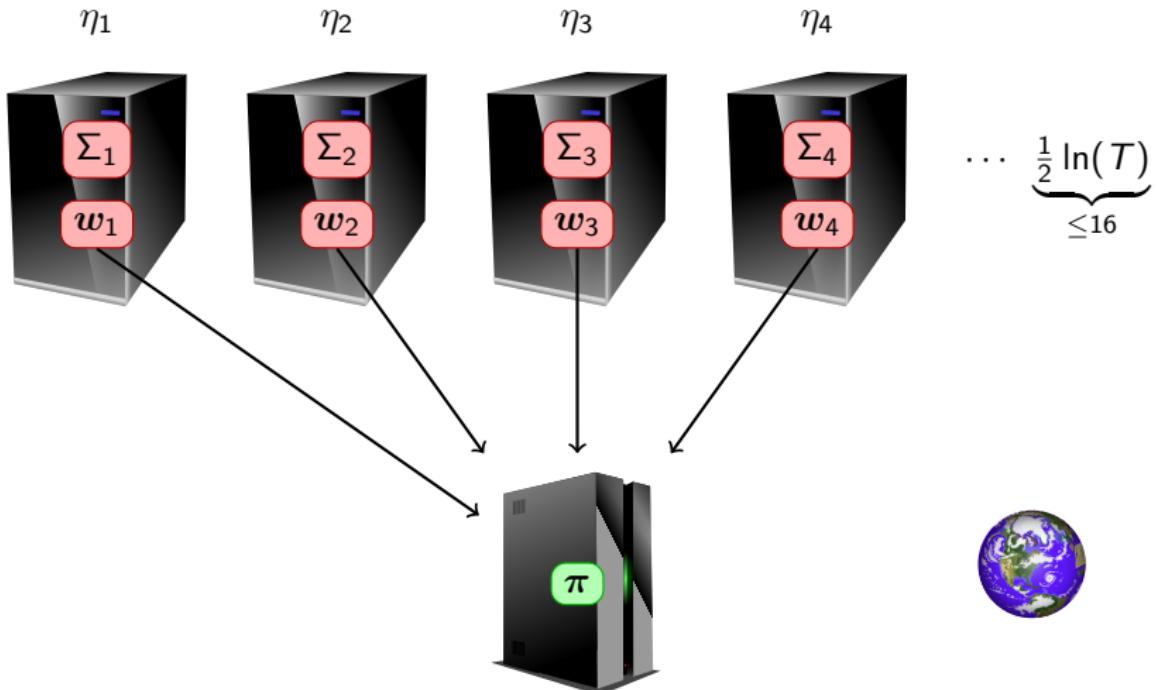
η_4



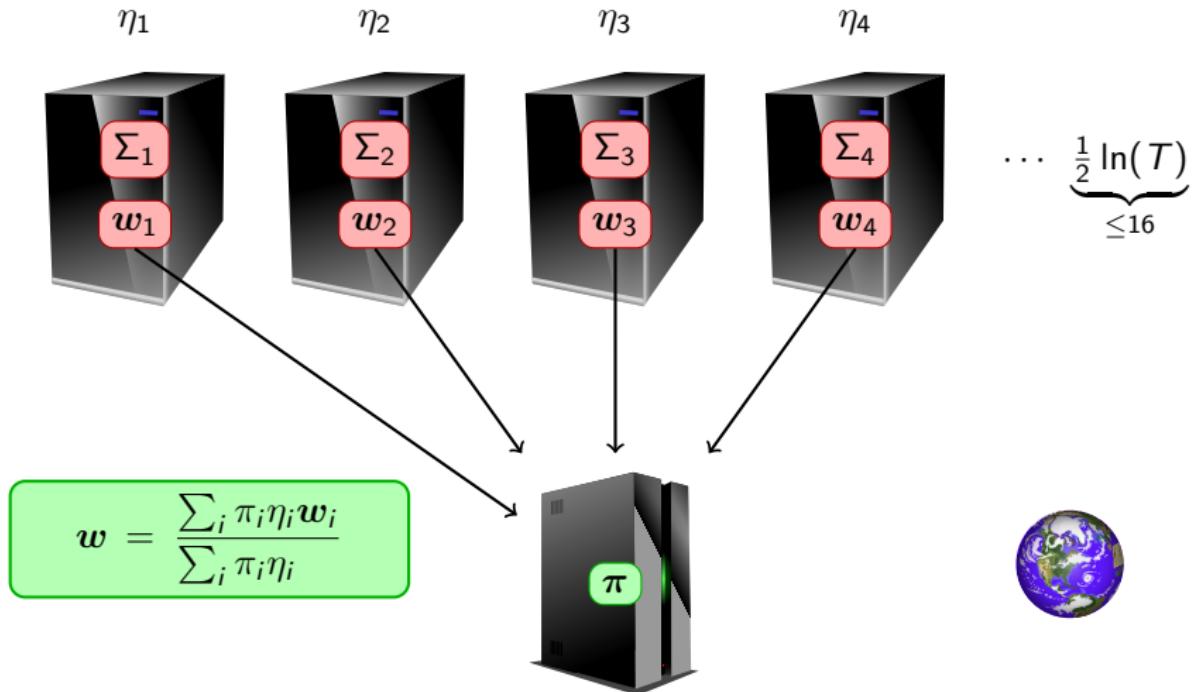
$$\dots \underbrace{\frac{1}{2} \ln(T)}_{\leq 16}$$



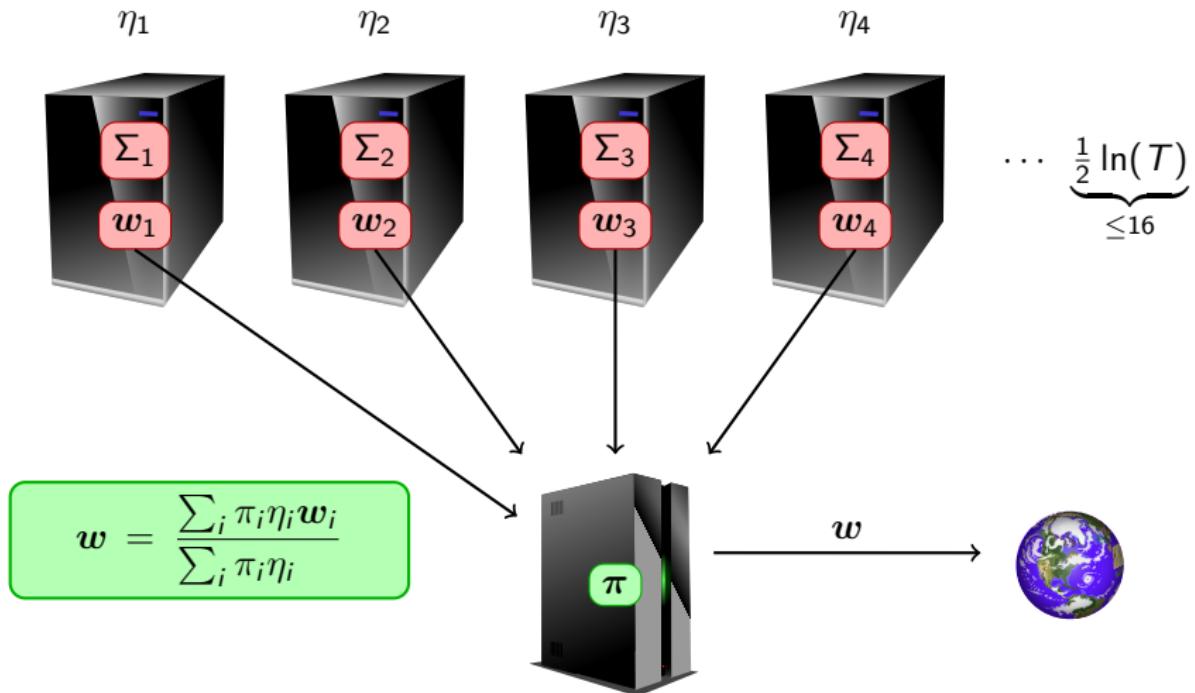
MetaGrad Algorithm



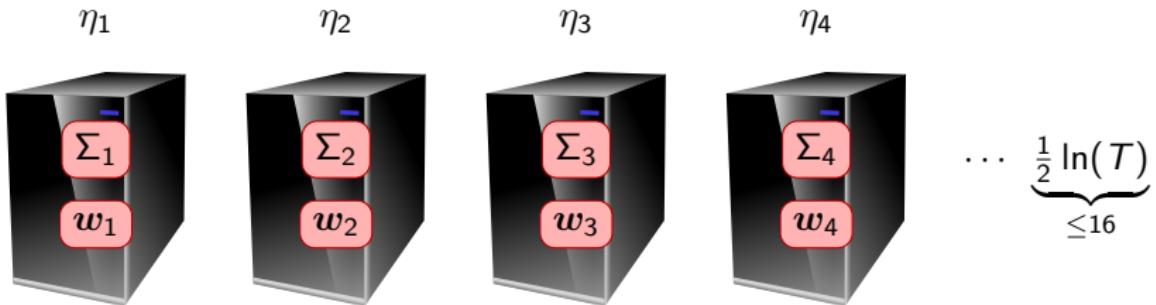
MetaGrad Algorithm



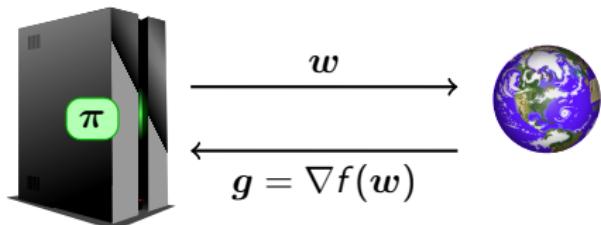
MetaGrad Algorithm



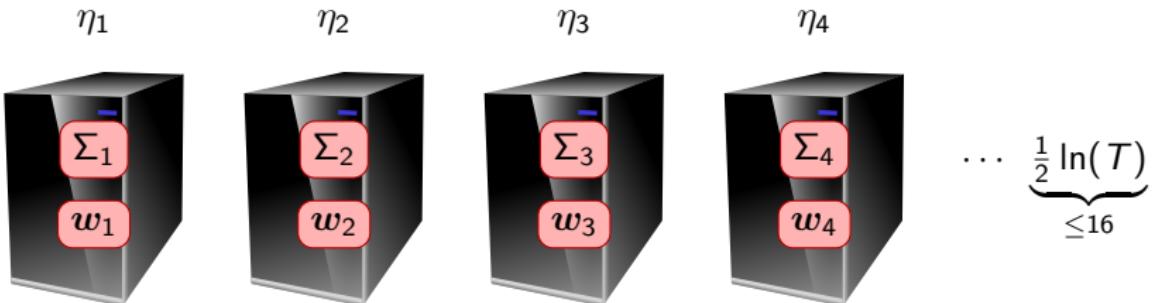
MetaGrad Algorithm



$$w = \frac{\sum_i \pi_i \eta_i w_i}{\sum_i \pi_i \eta_i}$$



MetaGrad Algorithm

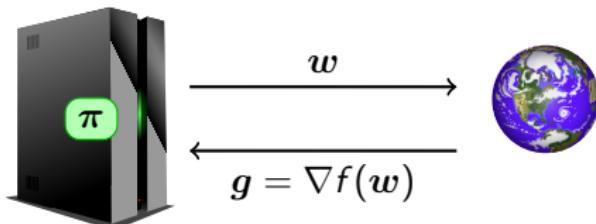


$$w = \frac{\sum_i \pi_i \eta_i w_i}{\sum_i \pi_i \eta_i}$$

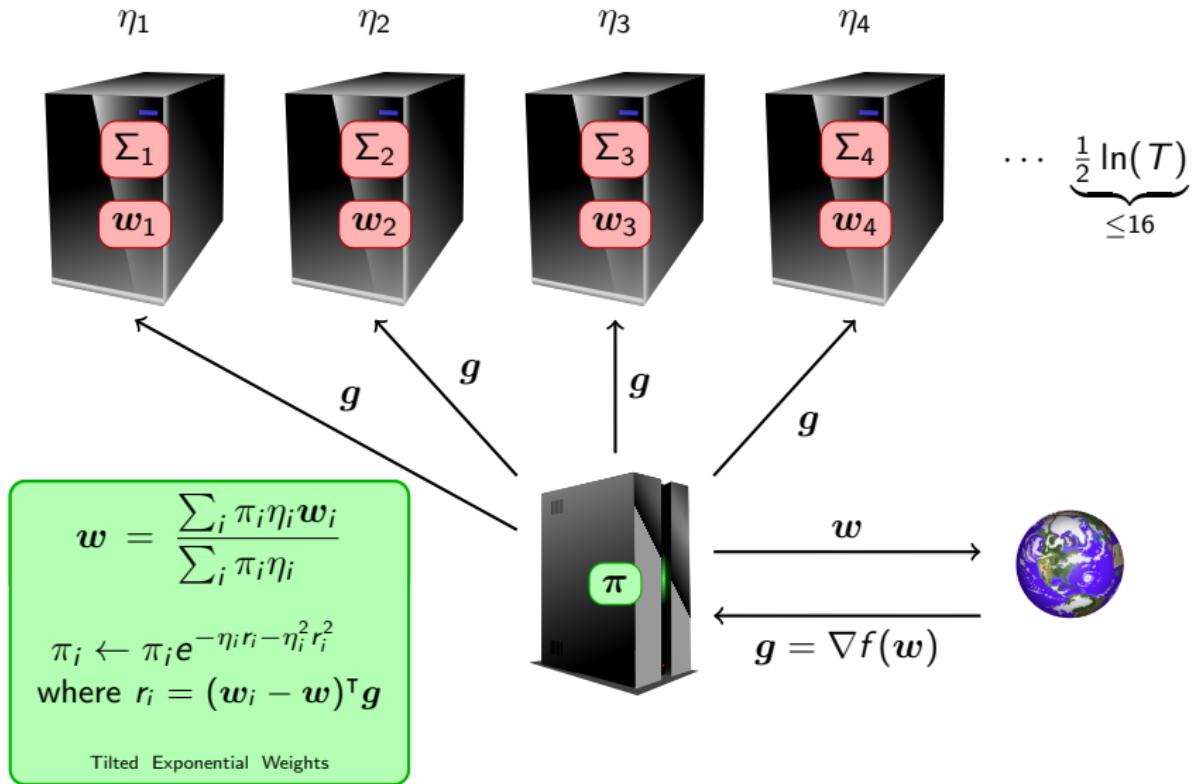
$$\pi_i \leftarrow \pi_i e^{-\eta_i r_i - \eta_i^2 r_i^2}$$

where $r_i = (w_i - w)^\top g$

Tilted Exponential Weights



MetaGrad Algorithm



MetaGrad Alg

$$\Sigma_i \leftarrow (\Sigma_i^{-1} + 2\eta_i^2 gg^\top)^{-1}$$

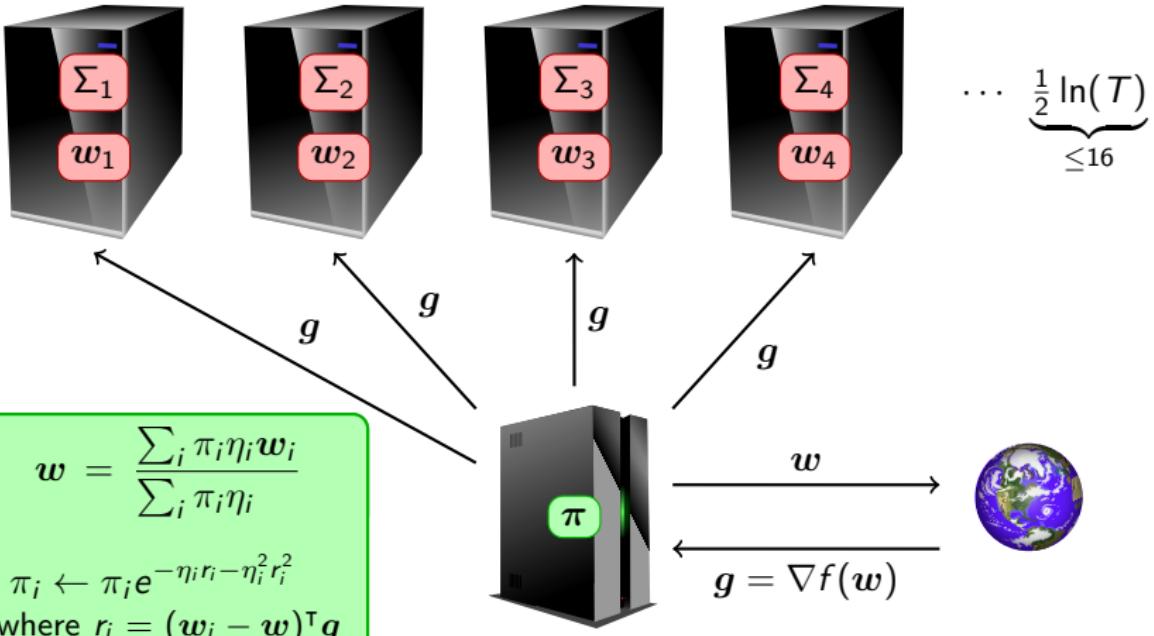
$$w_i \leftarrow w_i - \eta_i \Sigma_i g (1 + 2\eta_i r_i)$$

\approx Online Newton Step

η_1

η_2

η_3



Tilted Exponential Weights

MetaGrad: Multiple Eta Gradient Algorithm

Theorem (Van Erven, Koolen, 2016)

MetaGrad's Regret $\frac{u}{T}$ is bounded by

$$\sum_{t=1}^T (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t \preccurlyeq \begin{cases} \sqrt{T \ln \ln T} \\ \sqrt{\mathcal{V}_T^u d \ln T} + d \ln T \end{cases}$$

where

$$\mathcal{V}_T^u = \sum_{t=1}^T ((\mathbf{u} - \mathbf{w}_t)^\top \mathbf{g}_t)^2$$

- ▶ By convexity, $\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{u}) \leq (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t$.

MetaGrad: Multiple Eta Gradient Algorithm

Theorem (Van Erven, Koolen, 2016)

MetaGrad's Regret $_{\mathcal{T}}^{\mathbf{u}}$ is bounded by

$$\sum_{t=1}^T (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t \preccurlyeq \begin{cases} \sqrt{T \ln \ln T} \\ \sqrt{V_T^{\mathbf{u}} d \ln T} + d \ln T \end{cases}$$

where

$$V_T^{\mathbf{u}} = \sum_{t=1}^T ((\mathbf{u} - \mathbf{w}_t)^\top \mathbf{g}_t)^2 = \sum_{t=1}^T (\mathbf{u} - \mathbf{w}_t)^\top \mathbf{g}_t \mathbf{g}_t^\top (\mathbf{u} - \mathbf{w}_t).$$

- ▶ By convexity, $\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{u}) \leq (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t$.
- ▶ Covariance: $\mathbf{g}_t \mathbf{g}_t^\top \propto \mathbf{X}_t \mathbf{X}_t^\top$ when $\ell_t(\mathbf{w}) = h_t(\langle \mathbf{w}, \mathbf{X}_t \rangle)$
e.g. hinge, logistic, squared loss

MetaGrad: Multiple Eta Gradient Algorithm

Theorem (Van Erven, Koolen, 2016)

MetaGrad's Regret $\frac{u}{T}$ is bounded by

$$\sum_{t=1}^T (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t \preccurlyeq \begin{cases} \sqrt{T \ln \ln T} \\ \sqrt{\mathcal{V}_T^u d \ln T} + d \ln T \end{cases}$$

where

$$\mathcal{V}_T^u = \sum_{t=1}^T ((\mathbf{u} - \mathbf{w}_t)^\top \mathbf{g}_t)^2 = \sum_{t=1}^T (\mathbf{u} - \mathbf{w}_t)^\top \mathbf{g}_t \mathbf{g}_t^\top (\mathbf{u} - \mathbf{w}_t).$$

- ▶ By convexity, $\ell_t(\mathbf{w}_t) - \ell_t(\mathbf{u}) \leq (\mathbf{w}_t - \mathbf{u})^\top \mathbf{g}_t$.
- ▶ Covariance: $\mathbf{g}_t \mathbf{g}_t^\top \propto \mathbf{X}_t \mathbf{X}_t^\top$ when $\ell_t(\mathbf{w}) = h_t(\langle \mathbf{w}, \mathbf{X}_t \rangle)$
e.g. hinge, logistic, squared loss
- ▶ Optimal learning rate η depends on \mathcal{V}_T^u , but \mathbf{u} unknown!
Solution: aggregate **multiple learning rates**

Consequences

1. Non-stochastic adaptation:

Convex ℓ_t	$\sqrt{T \ln \ln T}$
Exp-concave ℓ_t	$d \ln T$
Fixed convex $\ell_t = \ell$	$d \ln T$

Consequences

1. Non-stochastic adaptation:

Convex ℓ_t	$\sqrt{T \ln \ln T}$
Exp-concave ℓ_t	$d \ln T$
Fixed convex $\ell_t = \ell$	$d \ln T$

2. Stochastic without curvature

Suppose ℓ_t i.i.d. with stochastic optimum $\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_\ell[\ell(\mathbf{u})]$.

Then expected regret $\mathbb{E}[\text{Regret}_T^{\mathbf{u}^*}]$:

Absolute loss* $\ell_t(w) = w - X_t $	$\ln T$
Hinge loss* $\max\{0, 1 - Y_t \langle \mathbf{w}, \mathbf{X}_t \rangle\}$	$d \ln T$
(B, β)-Bernstein	$(B d \ln T)^{1/(2-\beta)} T^{(1-\beta)/(2-\beta)}$

*Conditions apply

1. Directional Derivative Condition

Theorem

If there exist $a, b > 0$ such that all ℓ_t satisfy

$$\ell_t(\mathbf{u}) \geq \ell_t(\mathbf{w}) + a(\mathbf{u} - \mathbf{w})^\top \nabla \ell_t(\mathbf{w}) + b((\mathbf{u} - \mathbf{w})^\top \nabla \ell_t(\mathbf{w}))^2 \text{ for } \mathbf{w} \in \mathcal{U},$$

then $O(d \ln T)$ regret w.r.t. \mathbf{u} .

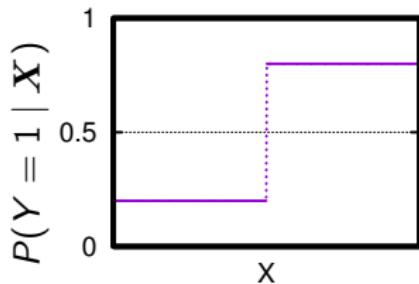
$a = 1$

- ▶ Satisfied by **exp-concave** functions [Hazan, Agarwal, and Kale, 2007]
- ▶ Requires quadratic curvature in direction of minimizer \mathbf{u} .

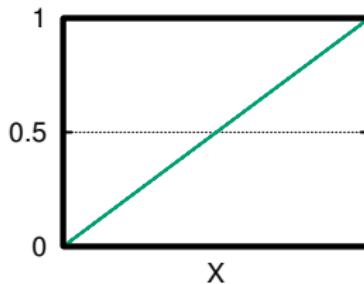
General a

- ▶ Satisfied for any **fixed convex** function $\ell_t = \ell$ with minimizer \mathbf{u} , even without any curvature, with $a = 2$ and $b = 1/(DG)$.

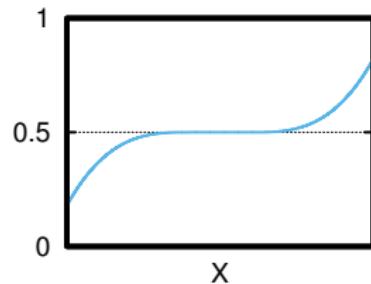
Related Work: Adaptivity to Stochastic Data in Batch Classification [Tsybakov, 2004]



easy
 $\beta = 1$

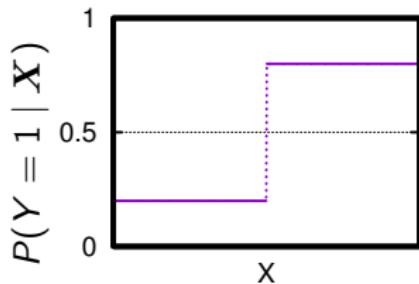


moderate
 $\beta = \frac{1}{2}$



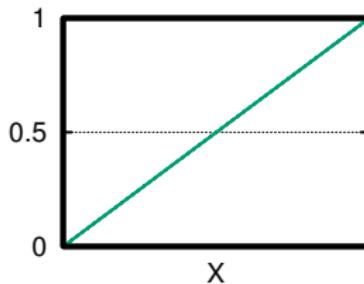
hard
 $\beta = 0$

Related Work: Adaptivity to Stochastic Data in Batch Classification [Tsybakov, 2004]



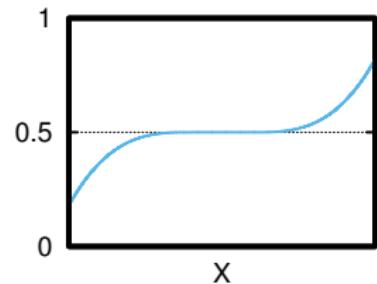
easy

$$\beta = 1$$



moderate

$$\beta = \frac{1}{2}$$



hard

$$\beta = 0$$

Definition $((B, \beta)$ -Bernstein Condition)

Losses are i.i.d. and

$$\mathbb{E}(\ell(\mathbf{w}) - \ell(\mathbf{u}^*))^2 \leq B(\mathbb{E}[\ell(\mathbf{w}) - \ell(\mathbf{u}^*)])^\beta \quad \text{for all } \mathbf{w},$$

where $\mathbf{u}^* = \arg \min_{\mathbf{u}} \mathbb{E}[\ell(\mathbf{u})]$ minimizes the expected loss.

2. Bernstein Condition for Online Learning

Suppose ℓ_t i.i.d. with stochastic optimum $\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\ell}[\ell(\mathbf{u})]$.

Standard Bernstein condition:

$$\mathbb{E} (\ell(\mathbf{w}) - \ell(\mathbf{u}^*))^2 \leq B (\mathbb{E} [\ell(\mathbf{w}) - \ell(\mathbf{u}^*)])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

2. Bernstein Condition for Online Learning

Suppose ℓ_t i.i.d. with stochastic optimum $\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\ell}[\ell(\mathbf{u})]$.

Standard Bernstein condition:

$$\mathbb{E} (\ell(\mathbf{w}) - \ell(\mathbf{u}^*))^2 \leq B (\mathbb{E} [\ell(\mathbf{w}) - \ell(\mathbf{u}^*)])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

Replace by **weaker linearized version**:

- ▶ Apply with $\tilde{\ell}(\mathbf{u}) = \langle \mathbf{u}, \nabla \ell(\mathbf{w}) \rangle$ instead of ℓ !
- ▶ By convexity, $\ell(\mathbf{w}) - \ell(\mathbf{u}^*) \leq \tilde{\ell}(\mathbf{w}) - \tilde{\ell}(\mathbf{u}^*)$.

$$\mathbb{E} ((\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w}))^2 \leq B (\mathbb{E} [(\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w})])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

2. Bernstein Condition for Online Learning

Suppose ℓ_t i.i.d. with stochastic optimum $\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\ell}[\ell(\mathbf{u})]$.

Standard Bernstein condition:

$$\mathbb{E} (\ell(\mathbf{w}) - \ell(\mathbf{u}^*))^2 \leq B (\mathbb{E} [\ell(\mathbf{w}) - \ell(\mathbf{u}^*)])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

Replace by **weaker linearized version**:

- ▶ Apply with $\tilde{\ell}(\mathbf{u}) = \langle \mathbf{u}, \nabla \ell(\mathbf{w}) \rangle$ instead of ℓ !
- ▶ By convexity, $\ell(\mathbf{w}) - \ell(\mathbf{u}^*) \leq \tilde{\ell}(\mathbf{w}) - \tilde{\ell}(\mathbf{u}^*)$.

$$\mathbb{E} ((\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w}))^2 \leq B (\mathbb{E} [(\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w})])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

Hinge loss (with $G = D = 1$): $\beta = 1$, $B = \frac{2\lambda_{\max}(\mathbb{E}[\mathbf{XX}^\top])}{\|\mathbb{E}[\mathbf{YX}]\|}$

2. Bernstein Condition for Online Learning

Suppose ℓ_t i.i.d. with stochastic optimum $\mathbf{u}^* = \arg \min_{\mathbf{u} \in \mathcal{U}} \mathbb{E}_{\ell}[\ell(\mathbf{u})]$.

Standard Bernstein condition:

$$\mathbb{E} (\ell(\mathbf{w}) - \ell(\mathbf{u}^*))^2 \leq B (\mathbb{E} [\ell(\mathbf{w}) - \ell(\mathbf{u}^*)])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

Replace by **weaker linearized version**:

- ▶ Apply with $\tilde{\ell}(\mathbf{u}) = \langle \mathbf{u}, \nabla \ell(\mathbf{w}) \rangle$ instead of ℓ !
- ▶ By convexity, $\ell(\mathbf{w}) - \ell(\mathbf{u}^*) \leq \tilde{\ell}(\mathbf{w}) - \tilde{\ell}(\mathbf{u}^*)$.

$$\mathbb{E} ((\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w}))^2 \leq B (\mathbb{E} [(\mathbf{w} - \mathbf{u}^*)^\top \nabla \ell(\mathbf{w})])^\beta \quad \text{for all } \mathbf{w} \in \mathcal{U}.$$

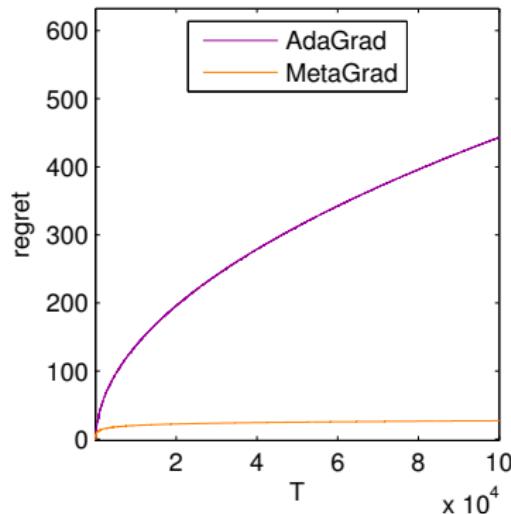
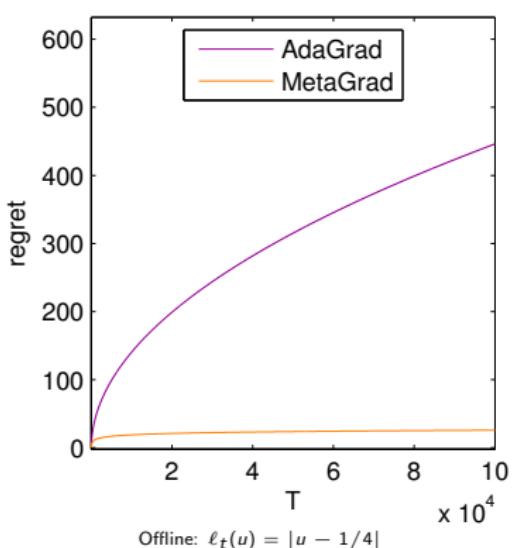
Hinge loss (with $G = D = 1$): $\beta = 1$, $B = \frac{2\lambda_{\max}(\mathbb{E}[\mathbf{XX}^\top])}{\|\mathbb{E}[\mathbf{YX}]\|}$

Theorem (Koolen, Grünwald, Van Erven, 2016)

$$\mathbb{E}[\text{Regret}_T^{\mathbf{u}^*}] \asymp (Bd \ln T)^{1/(2-\beta)} T^{(1-\beta)/(2-\beta)}$$

$$\text{Regret}_T^{\mathbf{u}^*} \asymp (Bd \ln T - \ln \delta)^{1/(2-\beta)} T^{(1-\beta)/(2-\beta)} \quad \text{w.p. } \geq 1 - \delta$$

Experiments



- ▶ MetaGrad: $O(\ln T)$ regret, AdaGrad: $O(\sqrt{T})$, match bounds
- ▶ Functions neither strongly convex nor smooth
- ▶ **Caveat:** comparison more complicated for higher dimensions, unless we run a separate copy of MetaGrad per dimension, like the diagonal version of AdaGrad runs GD per dimension

Summary

MetaGrad

- ▶ Consider **multiple learning rates** η simultaneously
- ▶ Learn η from the data, at very fast rate (pay only $\ln \ln T$)
- ▶ New adaptive variance bound

Variance bound implies fast rates in:

- ▶ all known cases: exp-concave, strong convex
- ▶ new cases with stochastic data, characterized by online version of Bernstein condition

Current/Future Work

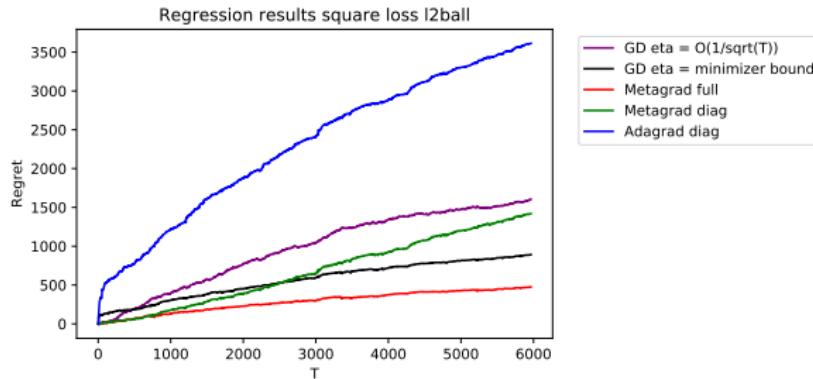
Computation

- ▶ Online learning often applied in high dimensions d
- ▶ Gradient descent: $O(d)$ work per step
- ▶ MetaGrad: $O(d^2)$ work per step + projection on domain
- ▶ Speed up MetaGrad to work in high dimensions by maintaining an **approximation of the matrix Σ**
(e.g. a low-dimensional sketch or only the diagonal)

Applications

- ▶ Football: predicting match winners, nr. of goals
- ▶ Deep learning
- ▶ ...

Preliminary Run on Football Data



Dirk van der Hoeven



Raphaël Deswarté

- ▶ Predict difference in goals in 6000 football games in English Premier League (Aug 2000–May 2017).
- ▶ Square loss on ball of radius 1.
- ▶ 37 features: running average of goals, shots on goal, shots over $m = 1, \dots, 10$ previous games; multiple ELO-like models; intercept.
- ▶ Data normalized: $\text{mean} = 0$, $\text{var} = 1$.

Papers

- ▶ T. van Erven and W. M. Koolen. **Metagrad: Multiple learning rates in online learning.** In Advances in Neural Information Processing Systems 29 (NIPS), pages 3666–3674, 2016.
- ▶ W. M. Koolen, P. Grünwald, and T. van Erven. **Combining adversarial guarantees and stochastic fast rates in online learning.** In Advances in Neural Information Processing Systems 29 (NIPS), pages 4457–4465, 2016.

References

- P. L. Bartlett, E. Hazan, and A. Rakhlin. Adaptive online gradient descent. In *Advances in Neural Information Processing Systems 20 (NIPS)*, pages 65–72, 2007.
- C. B. Do, Q. V. Le, and C.-S. Foo. Proximal regularization for online and batch learning. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 257–264, 2009.
- J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.
- E. Hazan. Introduction to online optimization. Draft, April 10, 2016, available from ocobook.cs.princeton.edu, 2016.
- E. Hazan and S. Kale. Extracting certainty from uncertainty: Regret bounded by variation in costs. *Machine learning*, 80(2-3):165–188, 2010.
- E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.
- F. Orabona. Simultaneous model selection and optimization through parameter-free stochastic learning. In *NIPS 27*, pages 1116–1124, 2014.
- F. Orabona and D. Pál. Coin betting and parameter-free online learning. In *NIPS 29*, 2016.
- F. Orabona, K. Crammer, and N. Cesa-Bianchi. A generalized online mirror descent with applications to classification and regression. *Machine Learning*, 99(3):411–435, 2015.
- A. B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004.