AdaGrad has regret larger than T

(and how to fix it)

Francesco Orabona **<u>Dávid Pál</u>**

Yahoo Research, New York

June 26, 2016

COLT 2016

Online Linear Optimization

Given convex set $K \subseteq \mathbb{R}^N$

For $t = 1, 2, \dots$

- predict $w_t \in K$
- receive loss vector $\ell_t \in \mathbb{R}^N$
- suffer loss $\langle \ell_t, w_t \rangle$

$$\mathrm{Regret}_T(u) = \underbrace{\sum_{t=1}^{T} \langle \ell_t, w_t \rangle}_{\text{algorithm's loss}} - \underbrace{\sum_{t=1}^{T} \langle \ell_t, u \rangle}_{\text{competitor's loss}}$$

We consider the **unbounded** domain $K = \mathbb{R}^N$

Gradient Descent

Gradient descent

$$w_1 = 0$$
$$w_{t+1} = w_t - \eta_t \ell_t$$

Popular step sizes:

1. $\eta_t = \frac{1}{\sqrt{t}}$ (Zinkevich)
2. $\eta_t = \frac{1}{\sqrt{\sum_{s=1}^{t} \|\ell_s\|_2^2}}$ (AdaGrad)

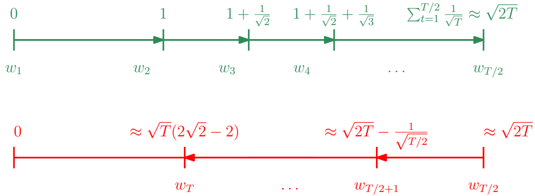
- Vowpal Wabbit, Spark MLlib, deep learning packages
- Both step sizes have regret $\Omega(T^{3/2})$ — this not a typo!

Regret is larger than T

One-dimensional loss vectors:

$$\underbrace{-1, -1, \ldots, -1}_{T/2}, \underbrace{+1, +1, \ldots, +1}_{T/2}$$

Zinkevich = AdaGrad: $\quad \eta_t = \frac{1}{\sqrt{\sum_{s=1}^{t} \|\ell_s\|_2^2}} = \frac{1}{\sqrt{t}}$



$$\text{Regret}_T(0) = \sum_{t=1}^{T} w_t \ell_t = -\sum_{t=1}^{T/2} w_t + \sum_{t=T/2+1}^{T} w_t \geq \frac{T^{3/2}}{20}$$

FTRL to the Rescue

Gradient Descent:

$$w_t = -\sum_{s=1}^{t-1} \eta_s \ell_s$$

FTRL (a.k.a. Dual Averaging):

$$w_t = -\eta_t \sum_{s=1}^{t-1} \ell_s$$

Theorem (Orabona-P. '15)

FTRL with $\eta_t = \frac{1}{\sqrt{\sum_{s=1}^{t-1} \|\ell_s\|_2^2}}$ *satisfies for all* $u \in \mathbb{R}^N$,

$$\text{Regret}_T(u) \leq \left(\frac{\|u\|_2^2}{2} + 2.75\right) \sqrt{\sum_{t=1}^{T} \|\ell_t\|_2^2} + 3.5\sqrt{T} \max_{t \leq T} \|\ell_t\|_2$$