A Simple yet Universal Strategy for Online Convex Optimization

Lijun Zhang[1] Guanghui Wang[1] Jinfeng Yi[2] Tianbao Yang[3]

[1]Nanjing University, China

[2]Frontis.AI, China

[3]The University of Iowa, USA

The 39th International Conference on Machine Learning (ICML 2022)

Outline

Outline

Online Convex Optimization [Zinkevich, 2003]

- The Learning Process

1: **for** $t = 1, 2, \ldots, T$ **do**

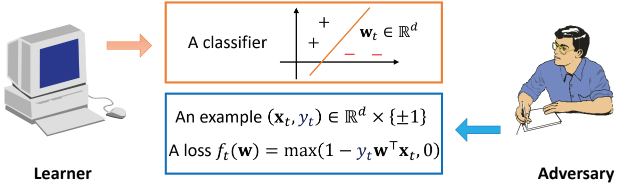
4: **end for**

Online Convex Optimization [Zinkevich, 2003]

- The Learning Process

for $t = 1, 2, \ldots, T$ **do**
  Learner picks a decision $\mathbf{x}_t$ from a convex set $\mathcal{X}$
  Adversary chooses a convex function $f_t(\cdot)$

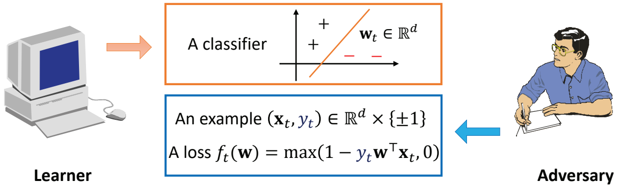
end for



Online Convex Optimization [Zinkevich, 2003]

The Learning Process

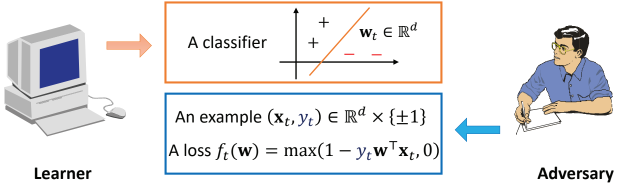
for $t = 1, 2, \ldots, T$ **do**
    Learner picks a decision $\mathbf{x}_t$ from a convex set $\mathcal{X}$
    Adversary chooses a convex function $f_t(\cdot)$
    Learner suffers loss $f_t(\mathbf{x}_t)$ and updates $\mathbf{x}_t$
end for



Online Convex Optimization [Zinkevich, 2003]

- The Learning Process

for $t = 1, 2, \ldots, T$ **do**
    Learner picks a decision $\mathbf{x}_t$ from a convex set $\mathcal{X}$
    Adversary chooses a convex function $f_t(\cdot)$
    Learner suffers loss $f_t(\mathbf{x}_t)$ and updates $\mathbf{x}_t$
end for



- Regret

$$\text{Regret} = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

Existing Regret Bounds

- Convex Functions [Zinkevich, 2003]
  - Online Gradient Descent (OGD)

$$\text{Regret} = O\left(\sqrt{T}\right)$$

Existing Regret Bounds

- Convex Functions [Zinkevich, 2003]
  - Online Gradient Descent (OGD)

$$\text{Regret} = O\left(\sqrt{T}\right)$$

- Strongly Convex Functions [Shalev-Shwartz et al., 2007]
  - The modulus of strong convexity λ is known
  - Online Gradient Descent (OGD)

$$\text{Regret} = O\left(\frac{\log T}{\lambda}\right)$$

Existing Regret Bounds

- Convex Functions [Zinkevich, 2003]
  - Online Gradient Descent (OGD)

$$\text{Regret} = O\left(\sqrt{T}\right)$$

- Strongly Convex Functions [Shalev-Shwartz et al., 2007]
  - The modulus of strong convexity λ is known
  - Online Gradient Descent (OGD)

$$\text{Regret} = O\left(\frac{\log T}{\lambda}\right)$$

- Exponentially Concave Functions [Hazan et al., 2007]
  - The modulus of exponential concavity α is known
  - Online Newton Step (ONS)

$$\text{Regret} = O\left(\frac{d \log T}{\alpha}\right)$$

Problem-dependent Regret Bounds I

- Small-loss Bounds

[Srebro et al., 2010, Orabona et al., 2012, Wang et al., 2020b]

- Convex, λ-strongly convex, α-exp-concave functions

$$\text{Regret} = O\left(\sqrt{L_T^*}\right), \quad O\left(\frac{1}{\lambda}\log L_T^*\right), \quad O\left(\frac{d}{\alpha}\log L_T^*\right)$$

where $L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$

Problem-dependent Regret Bounds I

Small-loss Bounds

[Srebro et al., 2010, Orabona et al., 2012, Wang et al., 2020b]

- Convex, λ-strongly convex, α-exp-concave functions

$$\text{Regret} = O\left(\sqrt{L_T^*}\right), \quad O\left(\frac{1}{\lambda}\log L_T^*\right), \quad O\left(\frac{d}{\alpha}\log L_T^*\right)$$

where $L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})$

- Reduce to the minimax rates in the worst case, but can be better when the problem is easy

Problem-dependent Regret Bounds I

Small-loss Bounds

[Srebro et al., 2010, Orabona et al., 2012, Wang et al., 2020b]

- Convex, λ-strongly convex, α-exp-concave functions

$$\text{Regret} = O\left(\sqrt{L_T^*}\right), \quad O\left(\frac{1}{\lambda}\log L_T^*\right), \quad O\left(\frac{d}{\alpha}\log L_T^*\right)$$

where $L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$

- Reduce to the minimax rates in the worst case, but can be better when the problem is easy

ADAGRAD [Duchi et al., 2010, Duchi et al., 2011]

- Convex, λ-strongly convex functions

$$\text{Regret} = O\left(\sum_{j=1}^{d} \|\mathbf{g}_{1:T,j}\|\right), \quad O\left(\frac{1}{\lambda}\sum_{j=1}^{d}\log\|\mathbf{g}_{1:T,j}\|\right)$$

- Can be better when the gradients are sparse

Problem-dependent Regret Bounds II

- RMSprop and SC-RMSProp [Tieleman and Hinton, 2012, Mukkamala and Hein, 2017]
- Adam [Kingma and Ba, 2015, Reddi et al., 2018]
- SAdam [Wang et al., 2020a]

Problem-dependent Regret Bounds II

- RMSprop and SC-RMSProp [Tieleman and Hinton, 2012, Mukkamala and Hein, 2017]
- Adam [Kingma and Ba, 2015, Reddi et al., 2018]
- SAdam [Wang et al., 2020a]
- Gradient-variation Bounds [Chiang et al., 2012, Yang et al., 2014, Mohri and Yang, 2016]
  - Convex, λ-strongly convex, α-exp-concave functions

$$\text{Regret} = O\left(\sqrt{V_T}\right), \quad O\left(\frac{1}{\lambda}\log V_T\right), \quad O\left(\frac{d}{\alpha}\log V_T\right)$$

where $V_T = \sum_{t=1}^{T} \max_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$

  - Can be better if the online functions evolve gradually

Our Contributions

- Limitations of Traditional Algorithms
  - The applicable algorithms depend on the type of functions
  - Their hyper-parameters depend on the moduli of strong convexity and exponential concavity

Our Contributions

Limitations of Traditional Algorithms

- The applicable algorithms depend on the type of functions
- Their hyper-parameters depend on the moduli of strong convexity and exponential concavity

A Simple yet Universal Strategy for OCO

- Handle multiple types of convex functions simultaneously
- For strongly convex functions and exp-concave functions, it inherits the (problem-dependent or independent) regret bounds of existing algorithms
- For general convex functions, it maintains the minimax optimality and also achieves a small-loss bound

Outline

Universal Algorithms I

- Adaptive Online Gradient Descent (AOGD) [Bartlett et al., 2008]
  - Interpolates between $O(\sqrt{T})$ regret of general convex functions and $O(\log T)$ regret of strongly convex functions

LAMDA
Learning And Mining from DatA

Universal Algorithms I

- Adaptive Online Gradient Descent (AOGD) [Bartlett et al., 2008]
  - Interpolates between $O(\sqrt{T})$ regret of general convex functions and $O(\log T)$ regret of strongly convex functions
  - It needs to know the modulus of strong convexity
  - It does not support exp-concave functions

Universal Algorithms I

- Adaptive Online Gradient Descent (AOGD) [Bartlett et al., 2008]
  - Interpolates between $O(\sqrt{T})$ regret of general convex functions and $O(\log T)$ regret of strongly convex functions
  - It needs to know the modulus of strong convexity
  - It does not support exp-concave functions

- MetaGrad [van Erven and Koolen, 2016]
  - $O(\log T)$ surrogate losses for exp-concave functions
$$\ell_{t,\eta}^{exp}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2[(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t]^2$$
  - $O(\frac{d}{\alpha} \log T)$ regret for α-exp-concave functions
  - $O(\sqrt{T \log \log T})$ regret bound for general convex functions

Universal Algorithms I

- Adaptive Online Gradient Descent (AOGD) [Bartlett et al., 2008]
  - Interpolates between $O(\sqrt{T})$ regret of general convex functions and $O(\log T)$ regret of strongly convex functions
  - It needs to know the modulus of strong convexity
  - It does not support exp-concave functions

- MetaGrad [van Erven and Koolen, 2016]
  - $O(\log T)$ surrogate losses for exp-concave functions
  $$\ell_{t,\eta}^{exp}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2[(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t]^2$$
  - $O(\frac{d}{\alpha}\log T)$ regret for α-exp-concave functions
  - $O(\sqrt{T \log \log T})$ regret bound for general convex functions
  - It does not support strongly convex functions explicitly

Universal Algorithms II

- Maler [Wang et al., 2019]
  - $O(\log T)$ surrogate losses for strongly convex functions
$$\ell_{t,\eta}^{str}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 \|\mathbf{x}_t - \mathbf{x}\|^2$$
  - 1 surrogate loss for general convex functions
$$\ell_{t,\eta}^{con}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 D^2$$
  - $O(\frac{1}{\lambda} \log T)$ regret for λ-strongly functions
  - $O(\frac{d}{\alpha} \log T)$ regret for α-exp-concave functions
  - $O(\sqrt{T})$ regret bound for general convex functions

Learning And Mining from DatA

Universal Algorithms II

- Maler [Wang et al., 2019]
  - $O(\log T)$ surrogate losses for strongly convex functions
  $$\ell_{t,\eta}^{str}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 \|\mathbf{x}_t - \mathbf{x}\|^2$$
  - 1 surrogate loss for general convex functions
  $$\ell_{t,\eta}^{con}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 D^2$$
  - $O(\frac{1}{\lambda}\log T)$ regret for λ-strongly functions
  - $O(\frac{d}{\alpha}\log T)$ regret for α-exp-concave functions
  - $O(\sqrt{T})$ regret bound for general convex functions
- UFO [Wang et al., 2020b]
  - $O(\log T)$ surrogate losses for strongly convex and smooth functions
  - 1 surrogate loss for convex and smooth functions

Universal Algorithms III

- UFO [Wang et al., 2020b]
  - Small-loss regret bounds for three types of convex and smooth functions

Universal Algorithms III

- UFO [Wang et al., 2020b]
  - Small-loss regret bounds for three types of convex and smooth functions

Limitations of State-of-the-art Universal Methods (MetaGrad, Maler, and UFO)

- Need to design one surrogate loss for each possible type of functions
- Cannot utilize existing online algorithms to exploit the structure of the problem instance
- Except the small-loss bound, it is unclear how to generate other problem-dependent regret bounds

Outline

Our Universal Strategy

- Follow the framework of “Learning with Expert Advice”
  - Construct a set of experts for each possible type of functions (discretizing continuous variables if necessary)
  - Deploy a meta-algorithm to aggregate their predictions

Our Universal Strategy

- Follow the framework of "Learning with Expert Advice"
  - Construct a set of experts for each possible type of functions (discretizing continuous variables if necessary)
  - Deploy a meta-algorithm to aggregate their predictions

- Novel Ideas
  - The experts process the original functions
  - The meta-algorithm uses linearized losses, and yields a second-order bound with excess losses

$$\sum_{t=1}^{T}(\ell_t - \ell_t^i) = O\left(\sqrt{\sum_{t=1}^{T}(\ell_t - \ell_t^i)^2}\right), \ \forall i$$

ℓ_t and ℓ_t^i are losses of S meta-algorithm and i-th expert

Motivations I

- Regret can be decomposed as

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

$$= \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)}_{:=\text{meta-regret}} + \underbrace{\sum_{t=1}^{T} f_t(\mathbf{u}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})}_{:=\text{expert-regret}}$$

Motivations I

- Regret can be decomposed as

$$
\begin{aligned}
&\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \\
=& \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)}_{:=\text{meta-regret}} + \underbrace{\sum_{t=1}^{T} f_t(\mathbf{u}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})}_{:=\text{expert-regret}}
\end{aligned}
$$

- Meta-regret of Strongly Convex Functions

$$
\begin{aligned}
\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t) \leq & \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u}_t \rangle - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2 \\
= & \sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t) \right) - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2
\end{aligned}
$$

where $l_t(\mathbf{x}) = \langle \nabla f_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle$

LAMDA Learning And Mining from DatA

Motivations I

- Regret can be decomposed as

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$
$$= \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)}_{:=\text{meta-regret}} + \underbrace{\sum_{t=1}^{T} f_t(\mathbf{u}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})}_{:=\text{expert-regret}}$$

- Meta-regret of Strongly Convex Functions

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t) \leq \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u}_t \rangle - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2$$
$$= \sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t) \right) - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2$$

A negative term appears when using linearized losses

Motivations II

- Consequence of the Second-order Bound of Excess Losses

$$\sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t)\right) = O\left(\sqrt{\sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t)\right)^2}\right)$$

Motivations II

- Consequence of the Second-order Bound of Excess Losses

$$\sum_{t=1}^{T}\left(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t)\right)=O\left(\sqrt{\sum_{t=1}^{T}\left(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t)\right)^2}\right)$$

$$=O\left(\sqrt{\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{u}_t\rangle^2}\right)=O\left(\sqrt{G^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{u}_t\|^2}\right)$$

Learning And Mining from DatA

Motivations II

- Consequence of the Second-order Bound of Excess Losses

$$\sum_{t=1}^{T}\left(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t)\right)=O\left(\sqrt{\sum_{t=1}^{T}\left(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t)\right)^2}\right)$$

$$=O\left(\sqrt{\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{u}_t\rangle^2}\right)=O\left(\sqrt{G^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{u}_t\|^2}\right)$$

$$=O\left(\frac{G^2}{\lambda}\right)+\frac{\lambda}{2}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{u}_t\|^2$$

LAMDA Learning And Mining from DatA

Motivations II

- Consequence of the Second-order Bound of Excess Losses

$$\sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t)\right) = O\left(\sqrt{\sum_{t=1}^{T} \left(l_t(\mathbf{x}_t) - l_t(\mathbf{u}_t)\right)^2}\right)$$

$$= O\left(\sqrt{\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u}_t \rangle^2}\right) = O\left(\sqrt{G^2 \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2}\right)$$

$$= O\left(\frac{G^2}{\lambda}\right) + \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}_t\|^2$$

- It can exploit the previous negative term

Motivations II

- Consequence of the Second-order Bound of Excess Losses

$$\sum_{t=1}^{T}(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t)) = O\left(\sqrt{\sum_{t=1}^{T}(l_t(\mathbf{x}_t)-l_t(\mathbf{u}_t))^2}\right)$$

$$= O\left(\sqrt{\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{u}_t\rangle^2}\right) = O\left(\sqrt{G^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{u}_t\|^2}\right)$$

$$= O\left(\frac{G^2}{\lambda}\right) + \frac{\lambda}{2}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{u}_t\|^2$$

  - It can exploit the previous negative term

- Meta-regret of Strongly Convex Functions

$$\sum_{t=1}^{T}f_t(\mathbf{x}_t)-\sum_{t=1}^{T}f_t(\mathbf{u}_t) = O\left(\frac{G^2}{\lambda}\right)$$

The Meta-algorithm—Adapt-ML-Prod [Gaillard et al., 2014]

- The loss of the i-th expert E^i

$$\ell_t^i = \frac{\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t^i - \mathbf{x}_t \rangle + GD}{2GD} \in [0, 1]$$

- The loss of the meta-algorithm $\ell_t = \sum p_t^i \ell_t^i = \frac{1}{2}$
- The weight of expert E^i

$$p_t^i = \frac{\eta_{t-1}^i w_{t-1}^i}{\sum_{j=1}^{|\mathcal{E}|} \eta_{t-1}^j w_{t-1}^j}$$

$$\eta_{t-1}^i = \min\left\{\frac{1}{2}, \sqrt{\frac{\ln |\mathcal{E}|}{1 + \sum_{s=1}^{t-1} (\ell_s - \ell_s^i)^2}}\right\}, \ t \geq 1,$$

$$w_{t-1}^i = \left(w_{t-2}^i \left(1 + \eta_{t-2}^i (\ell_{t-1} - \ell_{t-1}^i)\right)\right)^{\frac{\eta_{t-1}^i}{\eta_{t-2}^i}}, \ t \geq 2$$

- The prediction of the meta-algorithm $\mathbf{x}_t = \sum p_t^i \mathbf{x}_t^i$

Experts for Strongly Convex Functions

- Candidate Expert-algorithms $\mathcal{A}_{str}$
  - OGD for strongly convex functions (SC-OGD) [Shalev-Shwartz et al., 2007]
  - ADAGRAD for strongly convex functions [Duchi et al., 2010]
  - Online extra-gradient descent (OEGD) for strongly convex and smooth functions [Chiang et al., 2012]
  - SC-RMSProp [Mukkamala and Hein, 2017]
  - SAdam [Wang et al., 2020a]
  - S^2OGD for strongly convex and smooth functions [Wang et al., 2020b]

Experts for Strongly Convex Functions

- Candidate Expert-algorithms $\mathcal{A}_{str}$
  - OGD for strongly convex functions (SC-OGD) [Shalev-Shwartz et al., 2007]
  - ADAGRAD for strongly convex functions [Duchi et al., 2010]
  - Online extra-gradient descent (OEGD) for strongly convex and smooth functions [Chiang et al., 2012]
  - SC-RMSProp [Mukkamala and Hein, 2017]
  - SAdam [Wang et al., 2020a]
  - S^2OGD for strongly convex and smooth functions [Wang et al., 2020b]
- Candidate Moduli of Strong Convexity $\mathcal{P}_{str}$

$$\mathcal{P}_{str} = \left\{ \frac{1}{T}, \frac{2}{T}, \frac{2^2}{T}, \cdots, \frac{2^N}{T} \right\}, \quad N = \lceil \log_2 T \rceil$$

Theoretical Guarantee for Strongly Convex Functions

- Two Standard Assumptions [Zinkevich, 2003]
  - The gradients of all functions are bounded by G
  - The diameter of the domain $\mathcal{X}$ is bounded by D

Theoretical Guarantee for Strongly Convex Functions

- Two Standard Assumptions [Zinkevich, 2003]
  - The gradients of all functions are bounded by G
  - The diameter of the domain $\mathcal{X}$ is bounded by D

Theorem 1

Let $R(A, \widehat{\lambda})$ be the regret bound of expert $E(A, \widehat{\lambda})$. If the online functions are λ-strongly convex with $\lambda \in [1/T, 1]$, USC satisfies

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \min_{A \in \mathcal{A}_{str}} R(A, \widehat{\lambda}) + O\left(\frac{\log \log T}{\lambda}\right)$$

where $\widehat{\lambda} \in \mathcal{P}_{str}$, and $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$

- When both the domain and gradients are bounded, USC achieves the best of all worlds, up to an additive factor of $O(\log \log T)$

Theoretical Guarantee for Strongly Convex Functions

- An Additional Assumptions [Srebro et al., 2010]
  - All the online functions are nonnegative, and H-smooth

Corollary 2

If the online functions are λ-strongly convex with $\lambda \in [1/T, 1]$,

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \left(\frac{1}{\lambda}\left(\min(\log L_T^*, \log V_T) + \log\log T\right)\right)$$

$L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum f_t(\mathbf{x})$ *and* $V_T = \sum \max_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$

- We obtain the best of the small-loss bound and the gradient-variation bound

Theoretical Guarantee for Strongly Convex Functions

- An Additional Assumptions [Srebro et al., 2010]
  - All the online functions are nonnegative, and *H*-smooth

Corollary 2

If the online functions are λ-strongly convex with $\lambda \in [1/T, 1]$,

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \left(\frac{1}{\lambda} \left(\min(\log L_T^*, \log V_T) + \log \log T \right) \right)$$

$L_T^* = \min_{\mathbf{x} \in \mathcal{X}} \sum f_t(\mathbf{x})$ *and* $V_T = \sum \max_{\mathbf{x} \in \mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$

- We obtain the best of the small-loss bound and the gradient-variation bound

- The complexity is $O(\log T)$ per iteration
  - Create an expert $E(A, \widehat{\lambda})$ for each $A \in \mathcal{A}_{str}$ and $\widehat{\lambda} \in \mathcal{P}_{str}$

Experts for Exp-concave Functions

- Candidate Expert-algorithms $\mathcal{A}_{exp}$
  - Online Newton step (ONS) [Hazan et al., 2007]
  - ONS for exp-concave and smooth functions [Orabona et al., 2012]
  - OEGD for exp-concave and smooth functions [Chiang et al., 2012]

Experts for Exp-concave Functions

- Candidate Expert-algorithms $\mathcal{A}_{exp}$
  - Online Newton step (ONS) [Hazan et al., 2007]
  - ONS for exp-concave and smooth functions [Orabona et al., 2012]
  - OEGD for exp-concave and smooth functions [Chiang et al., 2012]

- Candidate Moduli of Exponential Concavity $\mathcal{P}_{exp}$

$$\mathcal{P}_{exp} = \left\{ \frac{1}{T}, \frac{2}{T}, \frac{2^2}{T}, \cdots, \frac{2^N}{T} \right\}, \ N = \lceil \log_2 T \rceil$$

Experts for Exp-concave Functions

- Candidate Expert-algorithms $\mathcal{A}_{exp}$
  - Online Newton step (ONS) [Hazan et al., 2007]
  - ONS for exp-concave and smooth functions [Orabona et al., 2012]
  - OEGD for exp-concave and smooth functions [Chiang et al., 2012]

- Candidate Moduli of Exponential Concavity $\mathcal{P}_{exp}$

$$\mathcal{P}_{exp} = \left\{ \frac{1}{T}, \frac{2}{T}, \frac{2^2}{T}, \cdots, \frac{2^N}{T} \right\}, \ N = \lceil \log_2 T \rceil$$

- The additional complexity is also $O(\log T)$ per iteration
  - Create an expert $E(A, \widehat{\alpha})$ for each $A \in \mathcal{A}_{exp}$ and $\widehat{\alpha} \in \mathcal{P}_{exp}$

Theoretical Guarantee for Exp-concave Functions

Theorem 3

If the online functions are α-exp-concave with $\alpha \in [1/T, 1]$,

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \min_{A\in\mathcal{A}_{exp}} R(A, \widehat{\alpha}) + O\left(\frac{\log\log T}{\alpha}\right)$$

where $\widehat{\alpha} \in \mathcal{P}_{exp}$, and $\widehat{\alpha} \leq \alpha \leq 2\widehat{\alpha}$

- USC also achieves the best of all worlds

Theoretical Guarantee for Exp-concave Functions

Theorem 3

If the online functions are α-exp-concave with $\alpha \in [1/T, 1]$,

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \min_{A\in\mathcal{A}_{exp}} R(A, \widehat{\alpha}) + O\left(\frac{\log\log T}{\alpha}\right)$$

where $\widehat{\alpha} \in \mathcal{P}_{exp}$, and $\widehat{\alpha} \leq \alpha \leq 2\widehat{\alpha}$

- USC also achieves the best of all worlds

Corollary 4

Under the additional assumption,

$$\mathrm{Regret} = O\left(\frac{1}{\alpha}\left(d\min(\log L_T^*, \log V_T) + \log\log T\right)\right)$$

$L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum f_t(\mathbf{x})$ *and* $V_T = \sum \max_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$

LAMDA Learning And Mining from DatA

Experts for General Convex Functions

- Candidate Expert-algorithms $\mathcal{A}_{con}$
  - OGD [Zinkevich, 2003]
  - ADAGRAD [Duchi et al., 2011]
  - OEGD for convex and smooth functions [Chiang et al., 2012]
  - RMSprop [Tieleman and Hinton, 2012]
  - ADADELTA [Zeiler, 2012]
  - Adam [Kingma and Ba, 2015]
  - AO-FTRL [Mohri and Yang, 2016]
  - SOGD [Zhang et al., 2019]

Experts for General Convex Functions

- Candidate Expert-algorithms $\mathcal{A}_{con}$
  - OGD [Zinkevich, 2003]
  - ADAGRAD [Duchi et al., 2011]
  - OEGD for convex and smooth functions [Chiang et al., 2012]
  - RMSprop [Tieleman and Hinton, 2012]
  - ADADELTA [Zeiler, 2012]
  - Adam [Kingma and Ba, 2015]
  - AO-FTRL [Mohri and Yang, 2016]
  - SOGD [Zhang et al., 2019]

- The additional complexity is $O(1)$ per iteration
  - Create an expert $E(A)$ for each $A \in \mathcal{A}_{con}$

Theoretical Guarantee for General Convex Functions

Theorem 5

Let $R(A)$ be the regret bound of expert $E(A)$. We have

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \min_{A\in\mathcal{A}_{con}} R(A) + \textit{second-order meta-regret}$$

$$= \min_{A\in\mathcal{A}_{con}} R(A) + O\left(\sqrt{T \log\log T}\right)$$

- Sum of the expert-regret and the meta-regret

Theoretical Guarantee for General Convex Functions

Theorem 5

Let $R(A)$ be the regret bound of expert $E(A)$. We have

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \min_{A\in\mathcal{A}_{con}} R(A) + \textit{second-order meta-regret}$$
$$= \min_{A\in\mathcal{A}_{con}} R(A) + O\left(\sqrt{T \log\log T}\right)$$

- Sum of the expert-regret and the meta-regret

Corollary 6

Under the additional assumption,

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = O\left(\sqrt{L_T^* \log\log T}\right)$$

where $L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum f_t(\mathbf{x})$

LAMDA Learning And Mining from DatA

Experimental Setting

- Online Linear Classification

$$f_t(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} \max\left\{0, 1 - y_t^{(i)}\mathbf{x}^\top \mathbf{w}_t^{(i)}\right\} + \frac{\lambda}{2}\|\mathbf{x}\|^2$$

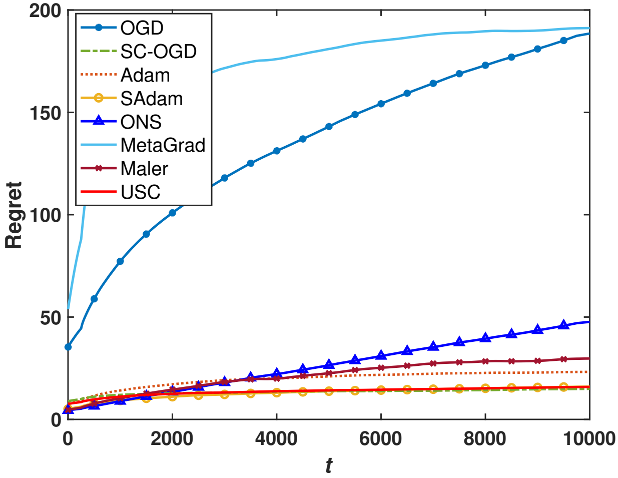
  - The a9a dataset [Chang and Lin, 2011]
  - Strongly convex functions ($\lambda = 0.02$) and general convex functions ($\lambda = 0$)

- Candidate Algorithms in USC
  - Strongly convex functions: SC-OGD and SAdam
  - Exp-concave functions: ONS [Hazan et al., 2007]
  - General convex functions: OGD and Adam

- Existing Universal Algorithms
  - MetaGrad [van Erven and Koolen, 2016] and Maler [Wang et al., 2019]

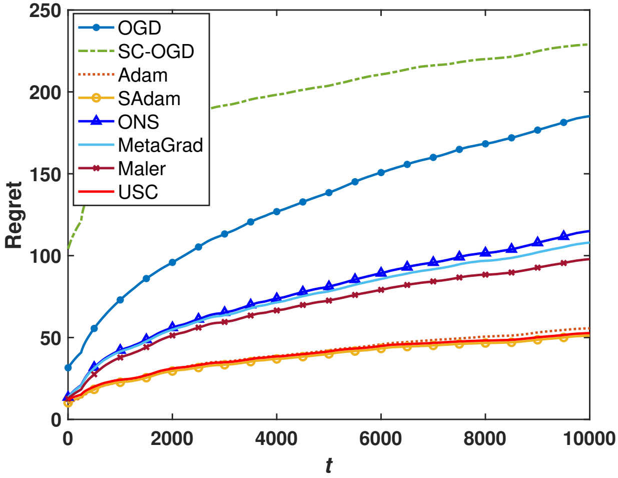
LAMDA Learning And Mining from DatA

Results for Strongly Convex Functions



- USC nearly match the best expert—SC-OGD
- USC is better than MetaGrad and Maler

Results for General Convex Functions



- USC nearly match the best expert—SAdam
- USC is better than MetaGrad and Maler

Outline

Conclusion and Future Work

- A Universal Strategy for OCO (USC)
  - The experts process the original functions, so that we can plug in any online solver as a black-box subroutine
  - The meta-algorithm uses linearized losses, and yields a second-order bound with excess losses

Conclusion and Future Work

- A Universal Strategy for OCO (USC)
  - The experts process the original functions, so that we can plug in any online solver as a black-box subroutine
  - The meta-algorithm uses linearized losses, and yields a second-order bound with excess losses
- Advantages of USC
  - Attains the best of all worlds for strongly convex functions and exp-concave functions
  - Attains a small-loss bound for general convex functions

Conclusion and Future Work

- A Universal Strategy for OCO (USC)
  - The experts process the original functions, so that we can plug in any online solver as a black-box subroutine
  - The meta-algorithm uses linearized losses, and yields a second-order bound with excess losses

- Advantages of USC
  - Attains the best of all worlds for strongly convex functions and exp-concave functions
  - Attains a small-loss bound for general convex functions

- Future Work
  - Extend to unbounded domains or gradients
  - Support dynamic regret and adaptive regret
  - Avoid fixing the value of the time horizon T

Reference I

Thanks!

Bartlett, P. L., Hazan, E., and Rakhlin, A. (2008).
Adaptive online gradient descent.
In *Advances in Neural Information Processing Systems 20*, pages 65–72.

Chang, C.-C. and Lin, C.-J. (2011).
LIBSVM: A library for support vector machines.
ACM Transactions on Intelligent Systems and Technology, 2(3):27:1–27:27.

Chiang, C.-K., Yang, T., Lee, C.-J., Mahdavi, M., Lu, C.-J., Jin, R., and Zhu, S. (2012).
Online optimization with gradual variations.
In *Proceedings of the 25th Annual Conference on Learning Theory*, pages 6.1–6.20.

Duchi, J., Hazan, E., and Singer, Y. (2010).
Adaptive subgradient methods for online learning and stochastic optimization.
In *Proceedings of the 23rd Annual Conference on Learning Theory*, pages 257–269.

Duchi, J., Hazan, E., and Singer, Y. (2011).
Adaptive subgradient methods for online learning and stochastic optimization.
Journal of Machine Learning Research, 12:2121–2159.

Gaillard, P., Stoltz, G., and van Erven, T. (2014).
A second-order bound with excess losses.
In *Proceedings of the 27th Conference on Learning Theory*, pages 176–196.

Reference II

Hazan, E., Agarwal, A., and Kale, S. (2007).
Logarithmic regret algorithms for online convex optimization.
Machine Learning, 69(2-3):169–192.

Kingma, D. P. and Ba, J. L. (2015).
Adam: A method for stochastic optimization.
In *International Conference on Learning Representations*.

Mohri, M. and Yang, S. (2016).
Accelerating online convex optimization via adaptive prediction.
In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 848–856.

Mukkamala, M. C. and Hein, M. (2017).
Variants of RMSProp and Adagrad with logarithmic regret bounds.
In *Proceedings of the 34th International Conference on Machine Learning*, pages 2545–2553.

Orabona, F., Cesa-Bianchi, N., and Gentile, C. (2012).
Beyond logarithmic bounds in online learning.
In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, pages 823–831.

Reddi, S. J., Kale, S., and Kumar, S. (2018).
On the convergence of Adam and beyond.
In *International Conference on Learning Representations*.

Shalev-Shwartz, S., Singer, Y., and Srebro, N. (2007).
Pegasos: primal estimated sub-gradient solver for SVM.
In *Proceedings of the 24th International Conference on Machine Learning*, pages 807–814.

Learning And Mining from DatA

Reference III

Srebro, N., Sridharan, K., and Tewari, A. (2010).
Smoothness, low-noise and fast rates.
In *Advances in Neural Information Processing Systems 23*, pages 2199–2207.

Tieleman, T. and Hinton, G. (2012).
Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude.
COURSERA: Neural networks for machine learning, pages 26–31.

van Erven, T. and Koolen, W. M. (2016).
MetaGrad: Multiple learning rates in online learning.
In *Advances in Neural Information Processing Systems 29*, pages 3666–3674.

Wang, G., Lu, S., Cheng, Q., Tu, W.-W., and Zhang, L. (2020a).
Sadam: A variant of Adam for strongly convex functions.
In *International Conference on Learning Representations*.

Wang, G., Lu, S., Hu, Y., and Zhang, L. (2020b).
Adapting to smoothness: A more universal algorithm for online convex optimization.
In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 6162–6169.

Wang, G., Lu, S., and Zhang, L. (2019).
Adaptivity and optimality: A universal algorithm for online convex optimization.
In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence*, pages 659–668.

Yang, T., Mahdavi, M., Jin, R., and Zhu, S. (2014).
Regret bounded by gradual variation for online convex optimization.
Machine Learning, 95:183–223.

Learning And Mining from DatA

Reference IV

Zeiler, M. D. (2012).
Adadelta: An adaptive learning rate method.
ArXiv e-prints, arXiv:1212.5701.

Zhang, L., Liu, T.-Y., and Zhou, Z.-H. (2019).
Adaptive regret of convex and smooth functions.
In *Proceedings of the 36th International Conference on Machine Learning*, pages 7414–7423.

Zinkevich, M. (2003).
Online convex programming and generalized infinitesimal gradient ascent.
In *Proceedings of the 20th International Conference on Machine Learning*, pages 928–936.