Efficient Online Learning for Dynamic Environments

Lijun Zhang

Dept. of Computer Science and Technology, Nanjing University

2018 International Workshop on Signal Processing, Optimization and Control (SPOC 2018)

Outline

Outline

LAMDA
Learning And Mining from DatA

What Happens in an Internet Minute?

http://www.dailyinfographic.com/what-happens-in-an-internet-minute-infographic

Outline

LAMDA
Learning And Mining from DatA

Online Learning [Shalev-Shwartz, 2011]

Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of answers to previous questions and possibly additional information.

Online Learning [Shalev-Shwartz, 2011]

Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of answers to previous questions and possibly additional information.

Online Learning

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

4: **end for**

Online Learning [Shalev-Shwartz, 2011]

Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of answers to previous questions and possibly additional information.

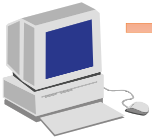
Online Learning

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

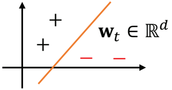
  Learner picks a decision $\mathbf{w}_t \in \mathcal{W}$

  Adversary chooses a function $f_t(\cdot)$

end for



A classifier



An example $(\mathbf{x}_t, y_t) \in \mathbb{R}^d \times \{\pm 1\}$

A loss $f_t(\mathbf{w}) = \max(1 - y_t \mathbf{w}^\top \mathbf{x}_t, 0)$

Learner

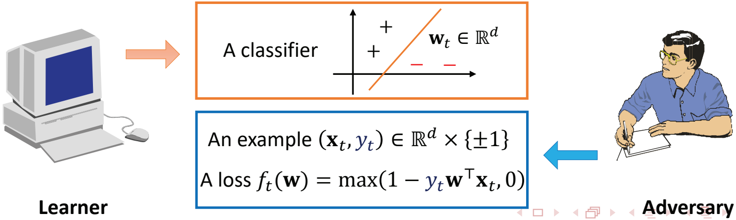
Adversary

Online Learning [Shalev-Shwartz, 2011]

Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of answers to previous questions and possibly additional information.

Online Learning

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Learner picks a decision $\mathbf{w}_t \in \mathcal{W}$
Adversary chooses a function $f_t(\cdot)$
3: Learner suffers loss $f_t(\mathbf{w}_t)$ and updates $\mathbf{w}_t$
4: **end for**



Online Learning [Shalev-Shwartz, 2011]

Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of answers to previous questions and possibly additional information.

Online Learning

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Learner picks a decision $\mathbf{w}_t \in \mathcal{W}$
Adversary chooses a function $f_t(\cdot)$
3: Learner suffers loss $f_t(\mathbf{w}_t)$ and updates $\mathbf{w}_t$
4: **end for**

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Outline

1. Introduction
   - Online Learning
   - Regret
2. Learning in Dynamic Environments
   - Adaptive Regret
   - Dynamic Regret
3. Our Work
   - Efficient Algorithms for Adaptive Regret
   - From Adaptive Regret to Dynamic Regret
4. Conclusion

LAMDA Learning And Mining from DatA

Regret

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Regret

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Regret

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

Regret

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Regret

$$\text{Regret} = \underbrace{\sum_{t=1}^{T} f_t(\mathbf{w}_t)}_{\text{Cumulative Loss of Online Learner}} - \underbrace{\min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})}_{\text{Minimal Loss of Offline Learner}}$$

Regret

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Regret

$$\text{Regret} = \underbrace{\sum_{t=1}^{T} f_t(\mathbf{w}_t)}_{\text{Cumulative Loss of Online Learner}} - \underbrace{\min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})}_{\text{Minimal Loss of Offline Learner}}$$

Hannan Consistent

$$\limsup_{T\to\infty} \frac{1}{T}\left(\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})\right) = 0, \text{ with probability } 1$$

LAMDA Learning And Mining from DatA

Regret

Cumulative Loss

$$\text{Cumulative Loss} = \sum_{t=1}^{T} f_t(\mathbf{w}_t)$$

Regret

$$\text{Regret} = \underbrace{\sum_{t=1}^{T} f_t(\mathbf{w}_t)}_{\text{Cumulative Loss of Online Learner}} - \underbrace{\min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})}_{\text{Minimal Loss of Offline Learner}}$$

Hannan Consistent

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) = o(T), \text{ with probability } 1$$

LAMDA Learning And Mining from DatA

Types of Online Learning

- Perceptron [Rosenblatt, 1958]
  - Online Classification
- Prediction with Expert Advice [Littlestone and Warmuth, 1994]
  - There are K experts in $\mathcal{W}$
  - As a meta-algorithm to combine different methods
- Online Convex Optimization [Zinkevich, 2003]
  - $f_1(\cdot), \ldots, f_T(\cdot)$ are convex
  - Online Classification, e.g., Online SVM
  - Online Regression, e.g., Online Least Squares

Online Gradient Descent (OGD)

- Algorithm

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Learner picks a decision $\mathbf{w}_t \in \mathcal{W}$
Adversary chooses a function $f_t(\cdot)$
3: Learner suffers loss $f_t(\mathbf{w}_t)$ and

$$\mathbf{w}_{t+1} = \Pi_{\mathcal{W}}\left(\mathbf{w}_t - \eta_t \nabla f_t(\mathbf{w}_t)\right)$$

4: **end for**

- The Projection Operator

$$\Pi_{\mathcal{W}}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{w} \in \mathcal{W}} \|\mathbf{w} - \mathbf{x}\|_2$$

Theoretical Guarantee

- Convex Functions [Zinkevich, 2003]

$$f_t(\mathbf{w}) \geq f_t(\mathbf{w}') + \langle \nabla f_t(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle, \ \forall \mathbf{w}, \mathbf{w}' \in \mathcal{W}$$

  - Online Gradient Descent with $\eta_t = 1/\sqrt{t}$

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) \leq \frac{D^2}{2}\sqrt{T} + \left(\sqrt{T} - \frac{1}{2}\right) G^2 = O\left(\sqrt{T}\right)$$

LAMDA Learning And Mining from DatA

Theoretical Guarantee

- Convex Functions [Zinkevich, 2003]

$$f_t(\mathbf{w}) \geq f_t(\mathbf{w}') + \langle \nabla f_t(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle, \ \forall \mathbf{w}, \mathbf{w}' \in \mathcal{W}$$

  - Online Gradient Descent with $\eta_t = 1/\sqrt{t}$

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) \leq \frac{D^2}{2}\sqrt{T} + \left(\sqrt{T} - \frac{1}{2}\right) G^2 = O\left(\sqrt{T}\right)$$

- Strongly Convex Functions [Hazan et al., 2007]

$$f_t(\mathbf{w}) \geq f_t(\mathbf{w}') + \langle \nabla f_t(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle + \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}'\|^2, \ \forall \mathbf{w}, \mathbf{w}' \in \mathcal{W}$$

  - Online Gradient Descent with $\eta_t = 1/(\lambda t)$

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) \leq \frac{G^2}{2\lambda}(1 + \log T) = O\left(\log T\right)$$

  - E.g., Online SVM [Shalev-Shwartz et al., 2007]

Exponentially Concave Functions

Definition

A function $f(\cdot) : \mathcal{W} \mapsto \mathbb{R}$ is α-exp-concave if $\exp(-\alpha f(\cdot))$ is concave over $\mathcal{W}$.

- For twice differentiable functions

$$\alpha \nabla f(\mathbf{w})[\nabla f(\mathbf{w})]^\top \preceq \nabla^2 f(\mathbf{w}), \ \forall \mathbf{w} \in \mathcal{W}.$$

- Examples
  - Logistic Loss for Classification

$$f(\mathbf{w}) = \log\left(1 + \exp(-y\mathbf{x}^\top \mathbf{w})\right)$$

  - Square Loss for Regression

$$f(\mathbf{w}) = (\mathbf{x}^\top \mathbf{w} - y)^2$$

  - Negative Logarithm Loss for Portfolio Management

$$f(\mathbf{w}) = -\log(\mathbf{x}^\top \mathbf{w})$$

Online Newton Step (ONS)

- Algorithm [Hazan et al., 2007]

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

2:

$$\mathbf{w}_{t+1} = \Pi_{\mathcal{W}}^{A_t}\left[\mathbf{w}_t - \frac{1}{\beta} A_t^{-1} \nabla f_t(\mathbf{w}_t)\right]$$

$$= \operatorname*{argmin}_{\mathbf{w} \in \mathcal{W}} \left(\mathbf{w} - \mathbf{w}'_{t+1}\right) A_t \left(\mathbf{w} - \mathbf{w}'_{t+1}\right)^\top$$

where

$$A_t = A_{t-1} + \nabla f_t(\mathbf{w}_t)[\nabla f_t(\mathbf{w}_t)]^\top, \ \mathbf{w}'_{t+1} = \mathbf{w}_t - \frac{1}{\beta} A_t^{-1} \nabla f_t(\mathbf{w}_t)$$

3: **end for**

- Regret

$$\sum_{t=1}^T f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{W}} \sum_{t=1}^T f_t(\mathbf{w}) \leq 5\left(\frac{1}{\alpha} + GD\right) d \log T = O\left(d \log T\right)$$

Outline

The Challenge

Regret → Static Regret

$$\mathrm{Regret} = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})$$
$$= \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{w}_*)$$

where $\mathbf{w}_* \in \mathrm{argmin}_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})$

- One of the decision is reasonably good during T rounds

The Challenge

Regret $\rightarrow$ Static Regret

$$\begin{aligned}\text{Regret} &= \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) \\ &= \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{w}_*)\end{aligned}$$

where $\mathbf{w}_* \in \operatorname{argmin}_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})$

- One of the decision is reasonably good during T rounds

Dynamic Environments

Different decisions will be good in different periods

- Recommendation: the interests of a user could change
- Stock market: the best stock changes over time

Outline

Adaptive Regret

Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

Adaptive Regret

Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

$$f_1(\cdot),\ f_2(\cdot), \ldots, f_\tau(\cdot), f_{\tau+1}(\cdot), \ldots, f_s(\cdot),\ f_{s+1}(\cdot), \ldots, f_{s+\tau-1}(\cdot), f_{s+\tau}(\cdot), \ldots$$

Adaptive Regret

Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

$$\underbrace{\boxed{f_1(\cdot), f_2(\cdot), \ldots, f_\tau(\cdot)}}_{\sum_{t=1}^{\tau} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{\tau} f_t(\mathbf{w})}, f_{\tau+1}(\cdot), \ldots, f_s(\cdot), f_{s+1}(\cdot), \ldots, f_{s+\tau-1}(\cdot), f_{s+\tau}(\cdot), \ldots$$

Adaptive Regret

- Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

$$\underbrace{f_1(\cdot), \overbrace{\boxed{f_2(\cdot), \ldots, f_\tau(\cdot)}_{}}^{}}_{\sum_{t=1}^{\tau} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{\tau} f_t(\mathbf{w})}\overbrace{\boxed{, f_{\tau+1}(\cdot)}}^{\sum_{t=2}^{\tau+1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=2}^{\tau+1} f_t(\mathbf{w})}, \ldots, f_s(\cdot), f_{s+1}(\cdot), \ldots, f_{s+\tau-1}(\cdot), f_{s+\tau}(\cdot), \ldots$$

Adaptive Regret

Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

$$\underbrace{f_1(\cdot), \overbrace{f_2(\cdot), \ldots, f_\tau(\cdot)}, f_{\tau+1}(\cdot)}^{\sum_{t=2}^{\tau+1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=2}^{\tau+1} f_t(\mathbf{w})}_{\sum_{t=1}^{\tau} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{\tau} f_t(\mathbf{w})}, \ldots, \underbrace{\boxed{f_s(\cdot), f_{s+1}(\cdot), \ldots, f_{s+\tau-1}(\cdot)}}_{\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w})}, f_{s+\tau}(\cdot), \ldots$$

Adaptive Regret

Adaptive Regret

[Hazan and Seshadhri, 2007, Daniely et al., 2015]

$$R(T,\tau) = \max_{[s,s+\tau-1]\subseteq[T]} \left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}) \right)$$

- Minimize the static regret over all intervals of length τ

$$\sum_{t=2}^{\tau+1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=2}^{\tau+1} f_t(\mathbf{w}) \qquad \sum_{t=s+1}^{s+\tau} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s+1}^{s+\tau} f_t(\mathbf{w})$$

$$f_1(\cdot), f_2(\cdot), \ldots, f_\tau(\cdot), f_{\tau+1}(\cdot), \ldots, f_s(\cdot), \boxed{f_{s+1}(\cdot), \ldots, f_{s+\tau-1}(\cdot), f_{s+\tau}(\cdot)}, \ldots$$

$$\sum_{t=1}^{\tau} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{\tau} f_t(\mathbf{w}) \qquad \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=s}^{s+\tau-1} f_t(\mathbf{w})$$

Adaptive Algorithms

- Following the Leading History

for $t = 1, 2, \ldots, T$ **do**
  Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$

end for

Adaptive Algorithms

- Following the Leading History

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

  Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$

  Initialize an expert E^t by running $\text{OGD}(f_t, \ldots)$

  Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$

end for

- Experts

$$
\begin{aligned}
E^1 &= \text{OGD}(f_1, f_2, f_3, \ldots, f_t, \ldots) \\
E^2 &= \text{OGD}(f_2, f_3, \ldots, f_t, \ldots) \\
E^3 &= \text{OGD}(f_3, \ldots, f_t, \ldots) \\
&\cdots \\
E^t &= \text{OGD}(f_t, \ldots)
\end{aligned}
$$

Adaptive Algorithms

- Following the Leading History

for $t = 1, 2, \ldots, T$ **do**
    Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
    Initialize an expert E^t by running OGD$(f_t, \ldots)$
    Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
    Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
end for

- Online Gradient Descent (OGD)

$$\mathbf{w}_{t+1}^i = \Pi_{\mathcal{W}}\left(\mathbf{w}_t^i - \eta_t \nabla f_t(\mathbf{w}_t^i)\right), \ \forall E^i \in \mathcal{S}_{t+1}$$

Adaptive Algorithms

- Following the Leading History

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
3: Initialize an expert E^t by running OGD$(f_t, \ldots)$
Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
4: Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
5: Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$

7: **end for**

- Prediction with Expert Advice
  - Exponential Weighting

$$\mathbf{w}_{t+1} = \sum_{E^i \in \mathcal{S}_{t+1}} p_{t+1}^i \mathbf{w}_{t+1}^i$$

$$p_{t+1}^i = p_t^i e^{-\alpha f_t(\mathbf{w}_t^i)}$$

LAMDA Learning And Mining from DatA

Adaptive Algorithms

- Following the Leading History

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
3: Initialize an expert E^t by running $\mathrm{OGD}(f_t, \ldots)$
Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
4: Get the prediction $\mathbf{w}^i_{t+1}$ for each expert $E^i \in \mathcal{S}_{t+1}$
5: Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}^i_{t+1} | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
6: Prune the set of experts $\mathcal{S}_{t+1}$
7: **end for**

- Experts

$$E^1 = \mathrm{OGD}(f_1, f_2, f_3, \ldots, f_t, \ldots)$$
$$\cancel{E^2 = \mathrm{OGD}(f_2, f_3, \ldots, f_t, \ldots)}$$
$$E^3 = \mathrm{OGD}(f_3, \ldots, f_t, \ldots)$$
$$\ldots$$
$$\cancel{E^t = \mathrm{OGD}(f_t, \ldots)}$$

Theoretical Guarantee

- Convex Functions [Jun et al., 2017]

$$R(T,\tau) = O\left(\sqrt{\tau \log T}\right)$$

- Strongly Convex Functions [Zhang et al., 2018]

$$R(T,\tau) = O\left(\log \tau \log T\right)$$

- Exponentially Concave Functions [Hazan and Seshadhri, 2007]

$$R(T,\tau) = O\left(d \log \tau \log T\right)$$

Outline

Dynamic Regret

- Static Regret

$$\text{Regret} = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{w}_*)$$

where $\mathbf{w}_* \in \operatorname{argmin}_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})$

- General Dynamic Regret

$$R(\mathbf{u}_1, \ldots, \mathbf{u}_T) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)$$

where $\mathbf{u}_1, \ldots, \mathbf{u}_T \in \mathcal{W}$

Dynamic Regret

- Static Regret

$$\text{Regret} = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{w}_*)$$

where $\mathbf{w}_* \in \mathrm{argmin}_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w})$

- General Dynamic Regret

$$R(\mathbf{u}_1, \ldots, \mathbf{u}_T) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)$$

where $\mathbf{u}_1, \ldots, \mathbf{u}_T \in \mathcal{W}$

- Worst-case Dynamic Regret

$$R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{w}_t^*)$$

where $\mathbf{w}_t^* \in \mathrm{argmin}_{\mathbf{w}\in\mathcal{W}} f_t(\mathbf{w})$

Worst-case Dynamic Regret

- The Challenge

Sublinear Dynamic Regret is Impossible in General!

Worst-case Dynamic Regret

- The Challenge

Sublinear Dynamic Regret is Impossible in General!

- Assumptions
  - Functional Variation [Besbes et al., 2015]

$$V_T := \sum_{t=2}^{T} \|f_t(\cdot) - f_{t-1}(\cdot)\|_\infty = \sum_{t=2}^{T} \max_{\mathbf{w}\in\mathcal{W}} |f_t(\mathbf{w}) - f_{t-1}(\mathbf{w})|$$

  - Path-length [Mokhtari et al., 2016]

$$P_T^* := \sum_{t=2}^{T} \|\mathbf{w}_t^* - \mathbf{w}_{t-1}^*\|$$

  - Squared Path-length [Zhang et al., 2017]

$$S_T^* := \sum_{t=2}^{T} \|\mathbf{w}_t^* - \mathbf{w}_{t-1}^*\|^2$$

A Representative Result

- Functional Variation [Besbes et al., 2015]
  - Restarted online gradient descent

$$R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) \leq \begin{cases} O\left(T^{2/3} V_T^{1/3}\right), & \text{Convex Functions} \\ O\left(\log T \sqrt{T V_T}\right), & \text{Strongly Convex Functions} \end{cases}$$

- Advantage
  - Dynamic regret is sublinear if V_T is sublinear

- Limitations
  - The algorithm needs to known an upper bound of V_T
  - It cannot be used in practice

LAMDA Learning And Mining from DatA

Outline

Our Recent Work I

Regret

1. Yuanyu Wan, Nan Wei, and **Lijun Zhang**. Efficient Adaptive Online Learning via Frequent Directions. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (**IJCAI**), 2018.
2. **Lijun Zhang**, Tianbao Yang, Rong Jin, Yichi Xiao, and Zhi-Hua Zhou. Online Stochastic Linear Optimization under One-bit Feedback. In *Proceedings of the 33rd International Conference on Machine Learning* (**ICML**), 2016.
3. **Lijun Zhang**, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. Online Bandit Learning for a Special Class of Non-convex Losses. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence* (**AAAI**), 2015.
4. **Lijun Zhang**, Jinfeng Yi, Rong Jin, Ming Lin, and Xiaofei He. Online Kernel Learning with a Near Optimal Sparsity Bound. In *Proceedings of the 30th International Conference on Machine Learning* (**ICML**), 2013.
5. **Lijun Zhang**, Rong Jin, Chun Chen, Jiajun Bu, and Xiaofei He. Efficient Online Learning for Large-Scale Sparse Kernel Logistic Regression. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence* (**AAAI**), 2012.

Learning And Mining from DatA

Our Recent Work II

Adaptive Regret

1. **Lijun Zhang**, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. Dynamic Regret of Strongly Adaptive Methods. In *Proceedings of the 35th International Conference on Machine Learning* (**ICML**), 2018.
2. Guanghui Wang, Dakuan Zhao, and **Lijun Zhang**. Minimizing Adaptive Regret with One Gradient per Iteration. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (**IJCAI**), 2018.

Dynamic Regret

1. **Lijun Zhang**, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. Improved Dynamic Regret for Non-degenerate Functions. In *Advances in Neural Information Processing Systems 30* (**NIPS**), 2017.
2. Tianbao Yang, **Lijun Zhang**, Rong Jin, and Jinfeng Yi. Tracking Slowly Moving Clairvoyant: Optimal Dynamic Regret of Online Learning with True and Noisy Gradient. In *Proceedings of the 33rd International Conference on Machine Learning* (**ICML**), 2016.

Our Recent Work II

Adaptive Regret

1. **Lijun Zhang**, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. Dynamic Regret of Strongly Adaptive Methods. In *Proceedings of the 35th International Conference on Machine Learning* (**ICML**), 2018.
2. Guanghui Wang, Dakuan Zhao, and **Lijun Zhang**. Minimizing Adaptive Regret with One Gradient per Iteration. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence* (**IJCAI**), 2018.

Dynamic Regret

1. **Lijun Zhang**, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. Improved Dynamic Regret for Non-degenerate Functions. In *Advances in Neural Information Processing Systems 30* (**NIPS**), 2017.
2. Tianbao Yang, **Lijun Zhang**, Rong Jin, and Jinfeng Yi. Tracking Slowly Moving Clairvoyant: Optimal Dynamic Regret of Online Learning with True and Noisy Gradient. In *Proceedings of the 33rd International Conference on Machine Learning* (**ICML**), 2016.

Outline

Adaptive Algorithms

- Following the Leading History

for $t = 1, 2, \ldots, T$ **do**
    Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
    Initialize an expert E^t by running OGD$(f_t, \ldots)$
    Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
    Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
    Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
    Prune the set of experts $\mathcal{S}_{t+1}$
end for

- Online Gradient Descent (OGD)

$$\mathbf{w}_{t+1}^i = \Pi_{\mathcal{W}} \left(\mathbf{w}_t^i - \eta_t \nabla f_t(\mathbf{w}_t^i) \right), \ \forall E^i \in \mathcal{S}_{t+1}$$

Adaptive Algorithms

- Following the Leading History

for $t = 1, 2, \ldots, T$ **do**
  Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
  Initialize an expert E^t by running OGD$(f_t, \ldots)$
  Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
  Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
  Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
  Prune the set of experts $\mathcal{S}_{t+1}$
end for

- Online Gradient Descent (OGD)

$$\mathbf{w}_{t+1}^i = \Pi_{\mathcal{W}}\left(\mathbf{w}_t^i - \eta_t \nabla f_t(\mathbf{w}_t^i)\right), \ \forall E^i \in \mathcal{S}_{t+1}$$

- Computational Cost per Iteration

$$|\mathcal{S}_{t+1}| \text{ gradient evaluations of } f_t(\cdot)$$

where $|\mathcal{S}_{t+1}| = O(\log t)$

Learning And Mining from DatA

Gradient Evaluations

- Nuclear-norm Regularized Losses

$$f_t(W) = \ell_t(W) + \lambda \|W\|_*$$

where $W \in \mathbb{R}^{m \times n}$

- Low-rank matrix regression
- Low-rank matrix approximation
- Low-rank multiclass classification

Gradient evaluations are expensive when m and n are large

- Mini-batch Losses

$$f_t(\mathbf{w}) = \frac{1}{k} \sum_{i=1}^{k} \ell(\mathbf{w}^\top \mathbf{x}_t^i, y_t^i)$$

Gradient evaluations are expensive when k is large

Online Learning with Surrogate Loss

- Following the Leading History [Wang et al., 2018]

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

  Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$

  Construct a surrogate loss $\ell_t(\cdot)$ from $\nabla f_t(\mathbf{w}_t)$

  Initialize an expert E^t by running OGD$(\ell_t, \ldots)$

  Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$

  Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$

  Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$

  Prune the set of experts $\mathcal{S}_{t+1}$

end for

Online Learning with Surrogate Loss

- Following the Leading History [Wang et al., 2018]

for $t = 1, 2, \ldots, T$ **do**
    Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
    Construct a surrogate loss $\ell_t(\cdot)$ from $\nabla f_t(\mathbf{w}_t)$
    Initialize an expert E^t by running OGD$(\ell_t, \ldots)$
    Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
    Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
    Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
    Prune the set of experts $\mathcal{S}_{t+1}$
end for

- Experts

$$
\begin{aligned}
E^1 &= \mathrm{OGD}(\ell_1, \ell_2, \ell_3, \ldots, \ell_t, \ldots) \\
E^2 &= \mathrm{OGD}(\ell_2, \ell_3, \ldots, \ell_t, \ldots) \\
&\ldots \\
E^t &= \mathrm{OGD}(\ell_t, \ldots)
\end{aligned}
$$

Online Learning with Surrogate Loss

- Following the Leading History [Wang et al., 2018]

for $t = 1, 2, \ldots, T$ **do**
    Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
    Construct a surrogate loss $\ell_t(\cdot)$ from $\nabla f_t(\mathbf{w}_t)$
    Initialize an expert E^t by running OGD$(\ell_t, \ldots)$
    Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
    Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
    Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
    Prune the set of experts $\mathcal{S}_{t+1}$
end for

- Online Gradient Descent (OGD)

$$\mathbf{w}_{t+1}^i = \Pi_{\mathcal{W}}\left(\mathbf{w}_t^i - \eta_t \nabla \ell_t(\mathbf{w}_t^i)\right), \ \forall E^i \in \mathcal{S}_{t+1}$$

Online Learning with Surrogate Loss

- Following the Leading History [Wang et al., 2018]

for $t = 1, 2, \ldots, T$ **do**
    Submit $\mathbf{w}_t \in \mathcal{W}$ and observes $f_t(\cdot)$
    Construct a surrogate loss $\ell_t(\cdot)$ from $\nabla f_t(\mathbf{w}_t)$
    Initialize an expert E^t by running OGD$(\ell_t, \ldots)$
    Add E^t to the set of experts $\mathcal{S}_{t+1} = \mathcal{S}_t \cup \{E^t\}$
    Get the prediction $\mathbf{w}_{t+1}^i$ for each expert $E^i \in \mathcal{S}_{t+1}$
    Predict $\mathbf{w}_{t+1}$ by combining $\{\mathbf{w}_{t+1}^i | \mathrm{E}^i \in \mathcal{S}_{t+1}\}$
    Prune the set of experts $\mathcal{S}_{t+1}$
end for

- Online Gradient Descent (OGD)

$$\mathbf{w}_{t+1}^i = \Pi_{\mathcal{W}}\left(\mathbf{w}_t^i - \eta_t \nabla \ell_t(\mathbf{w}_t^i)\right), \ \forall E^i \in \mathcal{S}_{t+1}$$

- Computational Cost per Iteration

$$1 \text{ gradient evaluation of } f_t(\cdot)$$

Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

which is convex

Learning And Mining from DatA

Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \le -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

which is convex

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \le \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle$$

which is convex

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

- Adaptive Regret

$$R(T, \tau) = O\left(\sqrt{\tau \log T}\right)$$

Strongly Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$

Strongly Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$

which is strongly convex

Strongly Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\lambda}{2} \|\mathbf{w} - \mathbf{w}_t\|^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\lambda}{2} \|\mathbf{w} - \mathbf{w}_t\|^2$$

which is strongly convex

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t) + \lambda(\mathbf{w}_t^i - \mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

Strongly Convex Functions

- First-order Condition

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$

which is strongly convex

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t) + \lambda(\mathbf{w}_t^i - \mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

- Adaptive Regret

$$R(T, \tau) = O\left(\log \tau \log T\right)$$

Exponentially Concave Functions

- First-order Condition [Hazan et al., 2007]

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\beta}{2} \left(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle \right)^2$$

Exponentially Concave Functions

- First-order Condition [Hazan et al., 2007]

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\beta}{2} \left(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle \right)^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\beta}{2} \left(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle \right)^2$$

which is exponentially concave

Exponentially Concave Functions

- First-order Condition [Hazan et al., 2007]

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\beta}{2} \left(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle \right)^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\beta}{2} \left(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle \right)^2$$

which is exponentially concave

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t) + \beta \langle \nabla f_t(\mathbf{w}_t), \mathbf{w}_t^i - \mathbf{w}_t \rangle \nabla f_t(\mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

Exponentially Concave Functions

- First-order Condition [Hazan et al., 2007]

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq -\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle - \frac{\beta}{2}(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle)^2$$

- Surrogate Loss

$$\ell_t(\mathbf{w}) = \langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle + \frac{\beta}{2}(\langle \nabla f_t(\mathbf{w}_t), \mathbf{w} - \mathbf{w}_t \rangle)^2$$

which is exponentially concave

- Properties
  - Gradient evaluation is easy

$$\nabla \ell_t(\mathbf{w}_t^i) = \nabla f_t(\mathbf{w}_t) + \beta \langle \nabla f_t(\mathbf{w}_t), \mathbf{w}_t^i - \mathbf{w}_t \rangle \nabla f_t(\mathbf{w}_t)$$

  - The regret bound is maintained

$$f_t(\mathbf{w}_t) - f_t(\mathbf{w}) \leq \ell_t(\mathbf{w}_t) - \ell_t(\mathbf{w})$$

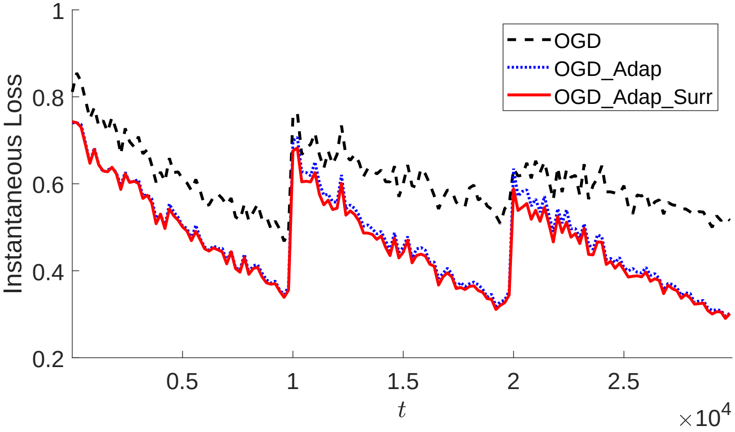
- Adaptive Regret

$$R(T, \tau) = O\left(d \log \tau \log T\right)$$

Experiments

Nuclear-norm Regularized Matrix Regression

$$f_t(W) = \frac{1}{2}\left(y_t - \mathrm{trace}(W^\top X_t)\right)^2 + b\|W\|_*$$

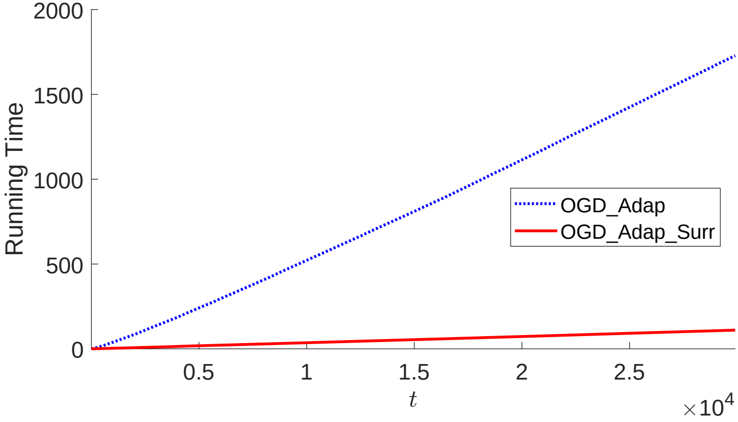


- $y_t = \mathrm{trace}(W_*^\top X_t) + \epsilon_t$, where W_* changes twice

Experiments

- Nuclear-norm Regularized Matrix Regression

$$f_t(W) = \frac{1}{2}\left(y_t - \mathrm{trace}(W^\top X_t)\right)^2 + b\|W\|_*$$

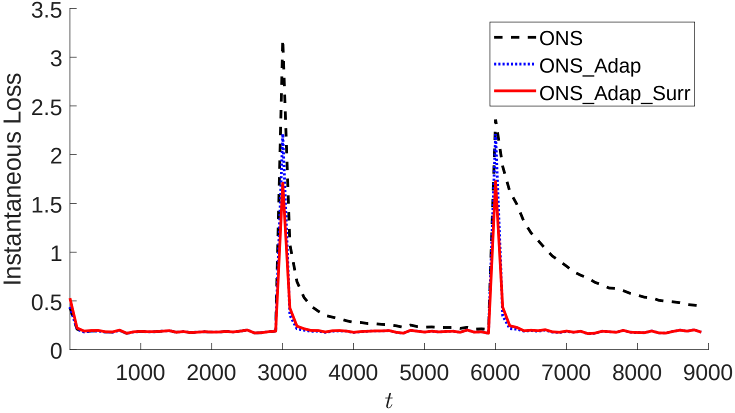


- $y_t = \mathrm{trace}(W_*^\top X_t) + \epsilon_t$, where W_* changes twice

Experiments

- Mini-batch Logistic Regression

$$f_t(\mathbf{w}) = \frac{1}{k} \sum_{i=1}^{k} \log \left(1 + \exp\left(-y_t^i \mathbf{w}^\top \mathbf{x}_t^i\right)\right)$$

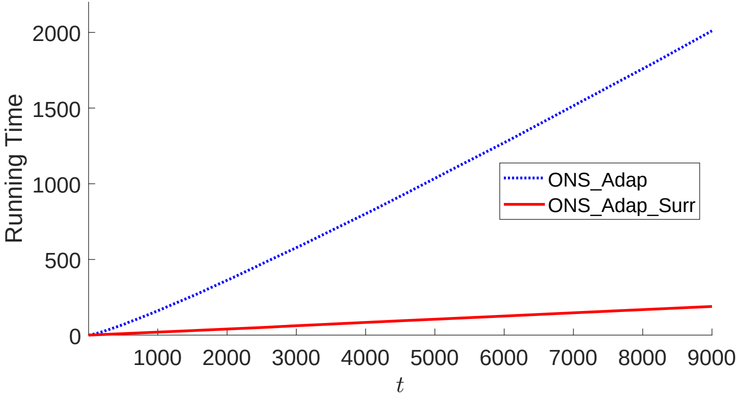


- the IJCNN01 dataset, where labels are flipped twice

Experiments

- Mini-batch Logistic Regression

$$f_t(\mathbf{w}) = \frac{1}{k} \sum_{i=1}^{k} \log\left(1 + \exp\left(-y_t^i \mathbf{w}^\top \mathbf{x}_t^i\right)\right)$$



- the IJCNN01 dataset, where labels are flipped twice

Outline

LAMDA Learning And Mining from DatA

Adaptive Regret versus Dynamic Regret

Adaptive Regret

$$R(T,\tau)=\max_{[s,s+\tau-1]\subseteq[T]}\left(\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w}_t)-\min_{\mathbf{w}\in\mathcal{W}}\sum_{t=s}^{s+\tau-1} f_t(\mathbf{w})\right)$$

- Well-studied

Worst-case Dynamic Regret

$$R(\mathbf{w}_1^*,\ldots,\mathbf{w}_T^*)=\sum_{t=1}^{T} f_t(\mathbf{w}_t)-\sum_{t=1}^{T} f_t(\mathbf{w}_t^*)=\sum_{t=1}^{T} f_t(\mathbf{w}_t)-\sum_{t=1}^{T}\min_{\mathbf{w}\in\mathcal{W}} f_t(\mathbf{w})$$

where $\mathbf{w}_t^*\in\operatorname{argmin}_{\mathbf{w}\in\mathcal{W}} f_t(\mathbf{w})$

- Partially addressed [Hall and Willett, 2013, Jadbabaie et al., 2015, Mokhtari et al., 2016, Yang et al., 2016, Zhang et al., 2017]

LAMDA Learning And Mining from DatA

From Adaptive Regret to Dynamic Regret

Theorem [Zhang et al., 2018]

- Let $\mathcal{I}_1 = [s_1, q_1], \mathcal{I}_2 = [s_2, q_2], \ldots, \mathcal{I}_k = [s_k, q_k]$ be a partition of $[1, T]$.
- Define the local functional variation of the i-th interval as

$$V_T(i) = \sum_{t=s_i+1}^{q_i} \max_{\mathbf{w}\in\mathcal{W}} |f_t(\mathbf{w}) - f_{t-1}(\mathbf{w})|$$

We have

$$R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) \leq \min_{\mathcal{I}_1,\ldots,\mathcal{I}_k} \sum_{i=1}^{k} \left(R(T, |\mathcal{I}_i|) + 2|\mathcal{I}_i| \cdot V_T(i)\right)$$

From Adaptive Regret to Dynamic Regret

Theorem [Zhang et al., 2018]

- Let $\mathcal{I}_1 = [s_1, q_1], \mathcal{I}_2 = [s_2, q_2], \ldots, \mathcal{I}_k = [s_k, q_k]$ be a partition of $[1, T]$.
- Define the local functional variation of the i-th interval as

$$V_T(i) = \sum_{t=s_i+1}^{q_i} \max_{\mathbf{w} \in \mathcal{W}} |f_t(\mathbf{w}) - f_{t-1}(\mathbf{w})|$$

We have

$$R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) \leq \min_{\mathcal{I}_1, \ldots, \mathcal{I}_k} \sum_{i=1}^{k} \left(R(T, |\mathcal{I}_i|) + 2|\mathcal{I}_i| \cdot V_T(i) \right)$$

Corollary

$$R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) \leq \min_{1 \leq \tau \leq T} \left(\frac{R(T, \tau)T}{\tau} + 2\tau V_T \right)$$

Dynamic Regret of Adaptive Algorithms

- Convex Functions

$$R(T,\tau) = O\left(\sqrt{\tau \log T}\right)$$

$$\Rightarrow R(\mathbf{w}_1^*,\ldots,\mathbf{w}_T^*) = O\left(\max\left\{\sqrt{T\log T}, T^{2/3}V_T^{1/3}\log^{1/3} T\right\}\right)$$

- Lower bound is $O(T^{2/3}V_T^{1/3})$ [Besbes et al., 2015]

Dynamic Regret of Adaptive Algorithms

- Convex Functions

$$R(T,\tau) = O\left(\sqrt{\tau \log T}\right)$$

$$\Rightarrow R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = O\left(\max\left\{\sqrt{T\log T}, T^{2/3}V_T^{1/3}\log^{1/3} T\right\}\right)$$

  - Lower bound is $O(T^{2/3}V_T^{1/3})$ [Besbes et al., 2015]

- Strongly Convex Functions

$$R(T,\tau) = O\left(\log \tau \log T\right)$$

$$\Rightarrow R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = O\left(\max\left\{\log T, \sqrt{TV_T \log T}\right\}\right)$$

  - Lower bound is $O(\sqrt{TV_T})$ [Besbes et al., 2015]

LAMDA Learning And Mining from DatA

Dynamic Regret of Adaptive Algorithms

- Convex Functions

$$R(T,\tau) = O\left(\sqrt{\tau \log T}\right)$$

$$\Rightarrow R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = O\left(\max\left\{\sqrt{T \log T}, T^{2/3} V_T^{1/3} \log^{1/3} T\right\}\right)$$

  - Lower bound is $O(T^{2/3} V_T^{1/3})$ [Besbes et al., 2015]

- Strongly Convex Functions

$$R(T,\tau) = O\left(\log \tau \log T\right)$$

$$\Rightarrow R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = O\left(\max\left\{\log T, \sqrt{T V_T \log T}\right\}\right)$$

  - Lower bound is $O(\sqrt{T V_T})$ [Besbes et al., 2015]

- Exponentially Concave Functions

$$R(T,\tau) = O\left(d \log \tau \log T\right)$$

$$\Rightarrow R(\mathbf{w}_1^*, \ldots, \mathbf{w}_T^*) = O\left(d \cdot \max\left\{\log T, \sqrt{T V_T \log T}\right\}\right)$$

LAMDA Learning And Mining from DatA

Our Contributions

1. Adaptive algorithms can be directly leveraged to minimize the dynamic regret.
2. The dynamic regrets are established without prior knowledge of the functional variation.
   - The results of [Besbes et al., 2015] needs to known an upper bound of V_T
3. This is the first time that exponential concavity is utilized in dynamic regret.

Outline

LAMDA
Learning And Mining from DatA

Conclusion and Future Work

Conclusion

- A brief introduction to adaptive regret and dynamic regret
- Efficient algorithms for adaptive regret
- From adaptive regret to dynamic regret

Future Work

- Adaptive regret that doesnot depend on T
- General dynamic regret

$$R(\mathbf{u}_1, \ldots, \mathbf{u}_T) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t)$$

  for any sequence $\mathbf{u}_1, \ldots, \mathbf{u}_T \in \mathcal{W}$
- Without convexity

Reference I

Thanks!

Besbes, O., Gur, Y., and Zeevi, A. (2015).
Non-stationary stochastic optimization.
Operations Research, 63(5):1227–1244.

Daniely, A., Gonen, A., and Shalev-Shwartz, S. (2015).
Strongly adaptive online learning.
In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1405–1411.

Hall, E. C. and Willett, R. M. (2013).
Dynamical models and tracking regret in online convex programming.
In *Proceedings of the 30th International Conference on Machine Learning*, pages 579–587.

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

Hazan, E. and Seshadhri, C. (2007).
Adaptive algorithms for online decision problems.
Electronic Colloquium on Computational Complexity, 88.

Jadbabaie, A., Rakhlin, A., Shahrampour, S., and Sridharan, K. (2015).
Online optimization: Competing with dynamic comparators.
In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, pages 398–406.

Jun, K.-S., Orabona, F., Wright, S., and Willett, R. (2017).
Improved Strongly Adaptive Online Learning using Coin Betting.
In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pages 943–951.

Reference II

Littlestone, N. and Warmuth, M. K. (1994).
The weighted majority algorithm.
Information and Computation, 108(2):212–261.

Mokhtari, A., Shahrampour, S., Jadbabaie, A., and Ribeiro, A. (2016).
Online optimization in dynamic environments: Improved regret rates for strongly convex problems.
In *IEEE 55th Conference on Decision and Control*, pages 7195–7201.

Rosenblatt, F. (1958).
The perceptron: a probabilistic model for information storage and organization in the brain.
Psychological Review, 65:386–407.

Shalev-Shwartz, S. (2011).
Online learning and online convex optimization.
Foundations and Trends in Machine Learning, 4(2):107–194.

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.

Wang, G., Zhao, D., and Zhang, L. (2018).
Minimizing adaptive regret with one gradient per iteration.
In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*.

Yang, T., Zhang, L., Jin, R., and Yi, J. (2016).
Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient.
In *Proceedings of the 33rd International Conference on Machine Learning*, pages 449–457.

LAMDA Learning And Mining from DatA

Reference III

Zhang, L., Yang, T., Jin, R., and Zhou, Z.-H. (2018).
Dynamic regret of strongly adaptive methods.
In *Proceedings of the 35th International Conference on Machine Learning*.

Zhang, L., Yang, T., Yi, J., Jin, R., and Zhou, Z.-H. (2017).
Improved dynamic regret for non-degenerate functions.
In *Advances in Neural Information Processing Systems 30*, pages 732–741.

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