Online Learning with Memory and Non-stochastic Control

Peng Zhao, Yu-Hu Yan, Yu-Xiang Wang, Zhi-Hua Zhou

Contact

{zhaop, yanyh, zhouzh}@lamda.nju.edu.cn

yuxiangw@cs.ucsb.edu

NANJING UNIVERSITY

LAMDA Learning And Mining from DatA

COMPUTER SCIENCE UC SANTA BARBARA *Computing. Reinvented.*

Online Learning to Online Decision Making

Standard Online Convex Optimization:

The loss of the t-th round is only related to the decision $\mathbf{w}_t$

Goal: to predict as well as the best offline decision

Online Convex Optimization with Memory

The loss of the t-th round can depend on the ***historical decisions***, for example, related to the past $m+1$ decision $\mathbf{w}_t, \mathbf{w}_{t-1}, \dots, \mathbf{w}_{t-m}$

⟹ *a simplified model to capture the memory effect in online decision making*

Policy regret: $$\text{Regret}_T = \sum_{t=1}^{T} f_t(\mathbf{w}_{t-m:t}) - \min_{\mathbf{v}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{v}, \dots, \mathbf{v})$$

Non-stationary Environments: online learning for real-world applications (such as whether forecasting, electricity prediction, etc)

→ optimal decision usually ***changes*** in non-stationary environments

Non-stationary OCO with Memory

Dynamic Policy Regret: competing with ***any*** comparators $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_T$

$$\text{D-Regret}_T(\mathbf{v}_{1:T}) = \sum_{t=1}^{T} f_t(\mathbf{w}_{t-m:t}) - \sum_{t=1}^{T} f_t(\mathbf{v}_{t-m:t})$$

adaptive to non-stationarity of environments

universal guarantee against any comparator sequence

specialize → *Static* policy regret of OCO w memory, when $\mathbf{v}_1 = \cdots = \mathbf{v}_T = \mathbf{v}^*$

$$\text{S-Regret}_T(\mathbf{v}^*) = \sum_{t=1}^{T} f_t(\mathbf{w}_{t-m:t}) - \sum_{t=1}^{T} f_t(\mathbf{v}^*, \dots, \mathbf{v}^*)$$

specialize → *Dynamic* regret of standard OCO, when memory length $m = 0$

$$\text{D-Regret}_T(\mathbf{v}_{1:T}) = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \sum_{t=1}^{T} f_t(\mathbf{v}_t)$$

Dynamic Regret Optimization

Reduction to OCO with switching cost: where $\tilde{f}_t(\mathbf{w}) \coloneqq f_t(\mathbf{w}, \dots, \mathbf{w})$

$$\text{D-Regret}_T(\mathbf{v}_{1:T}) \le \sum_{t=1}^{T} \tilde{f}_t(\mathbf{w}_t) - \sum_{t=1}^{T} \tilde{f}_t(\mathbf{v}_t) + \lambda \sum_{t=2}^{T} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2 + +\lambda \sum_{t=2}^{T} \|\mathbf{v}_t - \mathbf{v}_{t-1}\|_2$$

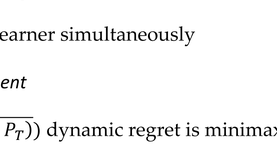
unary regret on $\tilde{f}_{1:T}$ — *switching cost* — *path-length*

Algorithm: Online Gradient Descent (OGD)

→ *by choosing the step size as $\mathcal{O}(\sqrt{(1+P_T)/T})$, OGD gives the optimal $\mathcal{O}(\sqrt{T(1+P_T)})$*

However, environmental non-stationarity P_T is usually unknown!

Online ensemble with **meta-base aggregation**: *Hedge the uncertainty!*



Step size pool: $\mathcal{H} = \left\{\eta_i \middle| \eta_i = 2^{i-1}\cdot\sqrt{\frac{D^2}{(\lambda G+G^2)T}}, i\in[N]\right\}$

$\text{D-Regret}_T(\mathbf{v}_{1:T}) \le \mathcal{O}(\sqrt{T(1+P_T)}) + \mathcal{O}(P_T) + \sum_{t=2}^{T}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2$

What about switching cost?

Scream: Switching-Cost Regularized Ensemble Algorithm for OCO with Memory

Switching Cost (key entity in OCO with memory):

Tension between dynamic regret and switching cost: optimizing dynamic regret needs the algorithm to ***move fast*** to catch up with the environment, while optimizing switching cost requires the algorithm to ***move slow***

$$\sum_{t=2}^{T}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2 \le D\sum_{t=2}^{T}\|\boldsymbol{p}_t - \boldsymbol{p}_{t-1}\|_1 + \sum_{t=2}^{T}\sum_{i=1}^{N} p_{t,i}\|\mathbf{w}_{t,i} - \mathbf{w}_{t-1,i}\|_2$$

maximum step size: $\eta_N = \mathcal{O}(1)$ ⇨ switching cost: $\|\mathbf{w}_{t,N} - \mathbf{w}_{t-1,N}\|_2 \le \mathcal{O}(\eta_N T) = \mathcal{O}(T)$! ⇨ *grows linearly in T!*

Algorithmically Enforce Low Switching Cost

Technical contributions: a novel *switching-cost-regularized surrogate loss*: $\ell_{t,i} \coloneqq \langle\nabla\tilde{f}_t(\mathbf{w}_t), \mathbf{w}_{t,i}\rangle + \lambda\|\mathbf{w}_{t,i} - \mathbf{w}_{t-1,i}\|_2$

→ *avoid directly controlling the switching cost but adding it as a penalty term into the loss function. Use algorithm to get low switching cost.*

meta regret: $\sum_{t=1}^{T}\langle\boldsymbol{p}_t, \boldsymbol{\ell}_t\rangle - \ell_{t,i} + \lambda D\sum_{t=2}^{T}\|\boldsymbol{p}_t - \boldsymbol{p}_{t-1}\|_1$ ⇨ Hedge: $p_{t+1,i} \propto p_{t,i}\exp(-\epsilon\ell_{t,i})$

base regret: $\sum_{t=1}^{T}\langle\nabla\tilde{f}_t(\mathbf{w}_t), \mathbf{w}_{t,i} - \mathbf{v}_t\rangle + \lambda\sum_{t=2}^{T}\|\mathbf{w}_{t,i} - \mathbf{w}_{t-1,i}\|_1$ ⇨ OGD: $\mathbf{w}_{t+1,i} = \Pi_{\mathcal{W}}[\mathbf{w}_{t,i} - \eta_i\nabla\tilde{f}_t(\mathbf{w}_t)]$

minimax optimal $\mathcal{O}(\sqrt{T(1+P_T)})$!

Optimal Memory Dependence:

$\lambda = \mathcal{O}(m^2)$ *is associated with memory length*

OCO with switching cost: $\sum_{t=1}^{T}\tilde{f}_t(\mathbf{w}_t) - \sum_{t=1}^{T}\tilde{f}_t(\mathbf{v}_t) + \lambda\sum_{t=2}^{T}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_2$ The best attained static regret: $\mathcal{O}(\sqrt{\lambda T})$ by OGD

Scream: $\mathcal{O}\left(\sqrt{\lambda T(1+P_T)} + \lambda^{3/4}\sqrt{T}\right) \Rightarrow \mathcal{O}\left(\lambda^{3/4}\sqrt{T}\right)$ regret in stationary environment ($P_T = 0$) ***Fails!***

Lazy Scream: ***slow down*** the update of base learners and meta learner simultaneously

→ $\mathcal{O}\left(\sqrt{\lambda T(1+P_T)} + \sqrt{\lambda T}\right) \Rightarrow \mathcal{O}\left(\sqrt{\lambda T}\right)$ *regret in stationary environment*

Optimality: we provide a lower bound to show that $\mathcal{O}(\sqrt{\lambda T(1+P_T)})$ dynamic regret is minimax optimal!

optimal memory dependence $\mathcal{O}(\sqrt{\lambda T(1+P_T)})$!

Application: Online Non-stochastic Control

Linear Dynamical System: $x_{t+1} = Ax_t + Bu_t + w_t$

Online Non-stochastic Control:

At each round $t = 1,2,\dots,T$

1. player observes a state x_t and provides a control u_t;
2. player suffers a convex loss $c_t(x_t, u_t)$;
3. environment chooses an *adversarial* noise w_t and evolves to state x_{t+1}.

Dynamic Policy Regret :

$$\text{D-Regret}_T(\pi_{1:T}) = \sum_{t=1}^{T} c_t(x_t, u_t) - \sum_{t=1}^{T} c_t\left(x_t^{\pi_t}, u_t^{\pi_t}\right)$$

competing with any controllers $\pi_1, \pi_2, \dots, \pi_T$

Reduction to OCO with Memory:

Policy parametrization:

Disturbance-Action Controller (DAC)

$u_t = -Kx_t + \sum_{i=1}^{H} M^{[i]}w_{t-i}$ where K and M are controller parameters

Truncation: under mild conditions, the states and actions that are more than m rounds before can be truncated at an acceptable cost

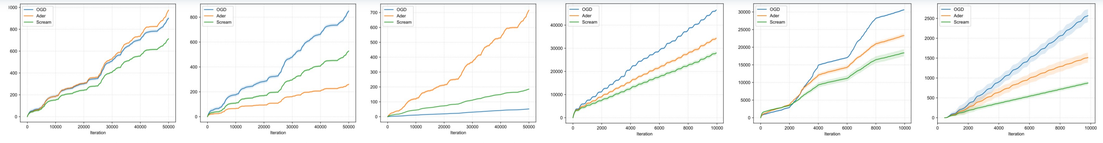
$\text{D-Regret}_T(M^*_{1:T}) = \sum_{t=1}^{T} f_t(M_{t-m:t}) - \sum_{t=1}^{T} f_t(M^*_{t-m:t})$

$M_{1:T}, M^*_{1:T}$: parameters of our controller and comparators $f_{1:T}$: truncated losses

Results: 1. the ***first*** controller with $\tilde{\mathcal{O}}(\sqrt{T(1+P_T)})$ ***dynamic policy regret***

2. extension to unknown system: $\tilde{\mathcal{O}}(\sqrt{T(1+P_T)} + T^{2/3})$ by system identification ("explore-then-commit ")

Experiments: OCO with Memory and Online Control



(d) overall loss ($\alpha = 1$) (e) cumulative loss ($\alpha = 1$) (f) switching cost ($\alpha = 1$) (a) LDS, gradual (b) LDS, abrupt (c) Inverted Pendulum

OCO with memory — Online non-stochastic control

P. Zhao, Y.-H. Yan, Y.-X. Wang, and Z.-H. Zhou. Online Learning with Memory and Non-stochastic Control. Short version published at AISTATS 2022, full version submitted to JMLR (available upon request).