Lecture 6. Online Convex Optimization

Advanced Optimization (Fall 2022)

Peng Zhao

zhaop@lamda.nju.edu.cn

Nanjing University

Outline

- Online Learning
- Online Convex Optimization
- Convex Functions
- Strongly Convex Functions
- Exp-concave Functions

A Brief Review of Statistical Learning

The fundamental goal of (supervised) learning: ***Risk Minimization (RM)***,

$$\min_{h \in \mathcal{H}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(h(\mathbf{x}), y)],$$

where

- h denotes the hypothesis (model) from the hypothesis space $\mathcal{H}$.
- $(\mathbf{x}, y)$ is an instance chosen from a unknown distribution $\mathcal{D}$.
- $\ell(h(\mathbf{x}), y)$ denotes the loss of using hypothesis h on the instance $(\mathbf{x}, y)$.

A Brief Review of Statistical Learning

Given a loss function f and distribution $\mathcal{D}$, the *expected risk* of predictor h is

$$R(h) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(h(\mathbf{x}), y)].$$

In practice, we can only access to a sample set $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$.
Thus, the following *empirical risk* is naturally defined:

$$\hat{R}_S(h) = \frac{1}{m}\sum_{i=1}^{m} \ell(h(\mathbf{x}_i), y_i).$$

A Brief Review of Statistical Learning

- A success story : characterization of sample complexity

Theorem 1 (Simple Generalization Bound). *Let $\mathcal{H}$ be a family of functions, with probablilty at least $1-\delta$, for any $h \in \mathcal{H}$, we have*

$$R(h) \leq \widehat{R}_S(h) + \sqrt{\frac{\log |\mathcal{H}|}{m}} + \sqrt{\frac{\log(1/\delta)}{2m}},$$

where $|\mathcal{H}|$ characterizes the complexity of $\mathcal{H}$.

Offline Towards Online Learning

- Traditional statistical machine learning: ***offline***

- Online learning scenario
  - In real applications, data are in the form of ***stream***
  - New data are being collected all the time: after observing a new data point, the model should be incrementally updated at a constant cost

A Formulation of Online Learning

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{w}_t \in \mathcal{W}$;

(2) and simultaneously environments pick an online function $f_t : \mathcal{W} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{w}_t)$, observes some information about f_t and updates the model.

Example (Online Classification): online function $f_t : \mathcal{W} \mapsto \mathbb{R}$ is composition of

(i) the loss $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \mapsto \mathbb{R}$, and

(ii) the hypothesis function $h : \mathcal{W} \times \mathcal{X} \mapsto \hat{\mathcal{Y}}$.

$$\Rightarrow \quad f_t(\mathbf{w}) = \ell(h(\mathbf{w}; \mathbf{x}_t), y_t) = \ell(\mathbf{w}^\top \mathbf{x}_t, y_t)$$ *for simplicity*

A Formulation of Online Learning

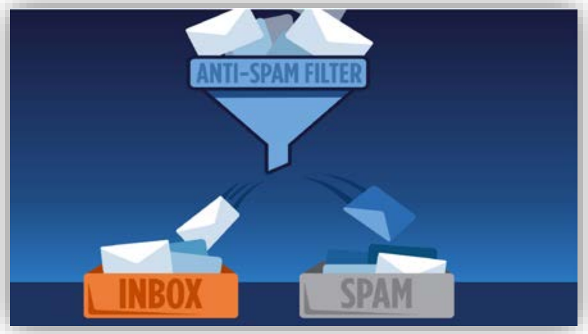
At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{w}_t \in \mathcal{W}$;

(2) and simultaneously environments pick an online function $f_t : \mathcal{W} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{w}_t)$, observes some information about f_t and updates the model.

Example: Spam filtering



(1) Player submits a spam classifier $\mathbf{w}_t$

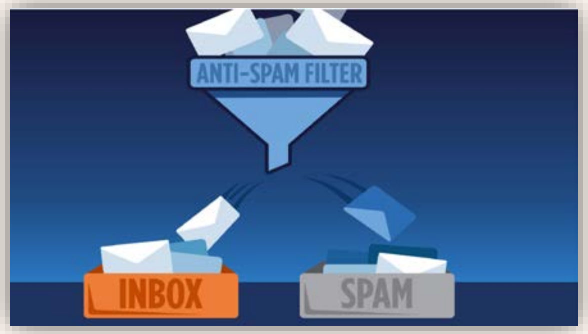
⇩

(2) A mail is revealed whether it is a spam

⇩

(3) Player suffers loss $f_t(\mathbf{w}_t)$ and updates model

Applications



spam detection (online classification/regression): At each time $t = 1, 2, \ldots$

- receive an email $\mathbf{x}_t \in \mathbb{R}^d$;
- predict whether it is a spam $\hat{y}_t \in \{-1, +1\}$;
- see its true label $y_t \in \{-1, +1\}$.

aggregating weather prediction (the expert problem): At each day $t = 1, 2, \ldots$

- obtain temperature predictions from N models;
- make the final prediction by randomly following a model according to the probability $p_t \in \Delta_N$;
- on the next day observe the loss of each model $f_t \in [0, 1]^N$.

Performance Measure

- Recall in the statistical learning: ***empirical risk***

$$\hat{R}(h) = \frac{1}{m}\sum_{i=1}^{m}\ell(h(\mathbf{x}_i), y_i).$$

- In online learning: ***sequential risk***

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) = \sum_{t=1}^{T}\ell\left(h(\mathbf{w}_t; \mathbf{x}_t), y_t\right).$$

meaning: cumulative loss of models trained on growing data stream $S_t = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_t, y_t)\}$.

Performance Measure

- In statistical learning, we use ***excess risk*** as measure for $\hat{h}$:

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(\hat{h}(\mathbf{x}),y)] - \inf_{h\in\mathcal{H}} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(h(\mathbf{x}),y)]$$

- In online learning, we define the following ***regret*** as measure:

$$\sum_{t=1}^{T} \ell(h(\mathbf{w}_t;\mathbf{x}_t),y_t) - \inf_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} \ell(h(\mathbf{w};\mathbf{x}_t),y_t)$$

cumulative loss of
the best solution in hindsight

Another View of Regret

- Ultimate goal: minimize the ***cumulative loss*** $\sum_{t=1}^{T} f_t(\mathbf{w}_t)$

- The cumulative loss highly depends on the loss function itself, so we need a benchmark:

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

- We hope the regret be sub-linear dependence with T.

$$\frac{\text{Regret}_T}{T} \to 0 \text{ as } T \to \infty$$

Hannan Consistency

ALT'16

Hannan Consistency in On-Line Learning in Case of Unbounded Losses Under Partial Monitoring*,**

Chamy Allenberg[1], Peter Auer[2], László Györfi[3], and György Ottucsák[3]

Compared with Statistical Learning

- Memory efficient

- Do not need i.i.d. assumption
  - the environment can be even adversarial
  - typically, the regret analysis does not need concentration

- Strictly harder than statistical learning
  - under non-i.i.d. assumption
  - online to batch conversion

Online-to-Batch Conversion

- An alternative way to solve statistic learning:
  - use the data in a sequential way
  - run any online algorithm
  - average the models returned

Algorithm 1 Online-to-Batch Conversion

Input: Data $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_T, y_T)\}$ i.i.d. sampled from the distribution $\mathcal{D}$, a bounded loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, 1]$, an online learning algorithm $\mathcal{A}$

for $i = 1, \cdots, T$ **do**

  let $\mathbf{w}_t$ be the output of algorithm $\mathcal{A}$ for this round

  Feed algorithm $\mathcal{A}$ with loss function $f_t(\mathbf{w}) = \ell(h(\mathbf{w}; \mathbf{x}_t), y_t)$

end for

return $\hat{\mathbf{w}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{w}_t$

Online-to-Batch Conversion

Theorem 2 (Online-to-Batch Conversion). *If the risk $R(\mathbf{w})$ is convex w.r.t. $\mathbf{w}$ with a bounded loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, 1]$, and the data $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_T, y_T)\}$ are i.i.d. sampled from the distribution $\mathcal{D}$, then with probability at least $1 - \delta$, the generalization error of the output of Algorithm 1 satisfies*

$$R(\hat{\mathbf{w}}) \leq R(\mathbf{w}^*) + \frac{\mathrm{Regret}_T}{T} + 2\sqrt{\frac{2\ln(2/\delta)}{T}}$$

where $R(\mathbf{w}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\ell(h(\mathbf{w};\mathbf{x}), y)$ is expected risk, $\mathbf{w}^* = \mathrm{argmin}_{\mathbf{w}\in\mathcal{W}} R(\mathbf{w})$ is bayes optimal classifier, $\mathrm{Regret}_T = \sum_{t=1}^T f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^T f_t(\mathbf{w})$ is the regret of $\mathcal{A}$ after T rounds.

Concentration Inequalities

Lemma 1 (Hoeffding's inequality). *Let* $X_1, \ldots, X_T \in [-B, B]$ *for some* $B > 0$ *be independent random variables such that* $\mathbb{E}[X_t] = 0$ *for all* $t \in [T]$*, then for all* $\delta \in (0, 1)$*,*

$$\Pr\left[\sum_{t=1}^{T} X_t \geq B\sqrt{2T \ln \frac{1}{\delta}}\right] \leq \delta$$

Lemma 2 (Azuma's inequality). *Let* $X_1, \ldots, X_T \in [-B, B]$ *for some* $B > 0$ *be a martingale difference sequence (i.e.,* $\forall t \in [T], \mathbb{E}[X_t \mid X_{t-1}, \ldots, X_1] = 0$ *), then* $\forall \delta > 0$*,*

$$\Pr\left[\sum_{t=1}^{T} X_t \geq B\sqrt{2T \ln \frac{1}{\delta}}\right] \leq \delta$$

Online-to-Batch Conversion

Theorem 2 (Online-to-Batch Conversion). *If the risk $R(\mathbf{w})$ is convex w.r.t. $\mathbf{w}$ with a bounded loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0,1]$, and the data $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_T, y_T)\}$ are i.i.d. sampled from the distribution $\mathcal{D}$, then with probability at least $1-\delta$, the generalization error of the output of Algorithm 1 satisfies*

$$R(\hat{\mathbf{w}}) \leq R(\mathbf{w}^*) + \frac{\mathrm{Regret}_T}{T} + 2\sqrt{\frac{2\ln(2/\delta)}{T}}$$

where $R(\mathbf{w}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} \ell(h(\mathbf{w};\mathbf{x}), y)$ *is expected risk,* $\mathbf{w}^* = \mathrm{argmin}_{\mathbf{w}\in\mathcal{W}} R(\mathbf{w})$ *is bayes optimal classifier,* $\mathrm{Regret}_T = \sum_{t=1}^T f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^T f_t(\mathbf{w})$ *is the regret of $\mathcal{A}$ after T rounds.*

Proof Sketch.

$$R(\hat{\mathbf{w}}) \underset{\text{Jensen's inequality}}{\leq} \frac{1}{T}\sum_{t=1}^T R(\mathbf{w}_t) \underset{\text{Azuma's inequality}}{\leq} \frac{1}{T}\sum_{t=1}^T f_t(\mathbf{w}_t) + \sqrt{\frac{2\ln(2/\delta)}{T}}$$

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

$$R(\mathbf{w}^*) + \sqrt{\frac{2\ln(2/\delta)}{T}} \underset{\text{Hoeffding's inequality}}{\geq} \frac{1}{T}\sum_{t=1}^T f_t(\mathbf{w}^\star) \geq \frac{1}{T}\min_{\mathbf{w}\in\mathcal{W}}\sum_{t=1}^T f_t(\mathbf{w})$$

Online-to-Batch Conversion

Proof.

$$
\begin{aligned}
R(\hat{\mathbf{w}}) &= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(h(\hat{\mathbf{w}};\mathbf{x}),y)\right] \\
&= \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\Big(h(\frac{1}{T}\sum_{t=1}^{T}\mathbf{w}_t;\mathbf{x}),y\Big)\right] \\
&\le \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(h(\mathbf{w}_t;\mathbf{x}),y)\right] && \text{(Jensen's inequality)} \\
&= \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}_t) + \sqrt{\frac{2\ln(2/\delta)}{T}} && \text{(Azuma's inequality with } X_t = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(h(\mathbf{w}_t;\mathbf{x}),y)] - f_t(\mathbf{w}_t))
\end{aligned}
$$

Online-to-Batch Conversion

Proof.

$$R(\hat{\mathbf{w}}) \leq \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}_t) + \sqrt{\frac{2\ln(2/\delta)}{T}}$$

$$= \min_{\mathbf{w}\in\mathcal{W}} \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}) + \frac{\text{Regret}_T}{T} + \sqrt{\frac{2\ln(2/\delta)}{T}} \quad \text{(definition of regret)}$$

$$\leq \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}^\star) + \frac{\text{Regret}_T}{T} + \sqrt{\frac{2\ln(2/\delta)}{T}}$$

$$\leq R(\mathbf{w}^\star) + \frac{\text{Regret}_T}{T} + 2\sqrt{\frac{2\ln(2/\delta)}{T}}$$

(Hoeffding's inequality with $X_t = f_t(\mathbf{w}^\star) - R(\mathbf{w}^\star)$)

□

A Trackable Case: **Online Convex Optimization**

- In general, the online learning formulation is ***hard*** to solve.

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{w}_t \in \mathcal{W}$;

(2) and simultaneously environments pick an online function $f_t : \mathcal{W} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{w}_t)$, observes some information about f_t and updates the model.

A Trackable Case: **Online Convex Optimization**

- In general, the online learning formulation is ***hard*** to solve.

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{w}_t \in \mathcal{W}$;

(2) and simultaneously environments pick an online function $f_t : \mathcal{W} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{w}_t)$, observes some information about f_t and updates the model.

⇨ ***Requiring feasible domain and online functions to be convex.***

Online Convex Optimization

- Online convex optimization framework
  - feasible domain is a convex set
  - online functions are convex

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t$ from a convex set $\mathcal{X} \subseteq \mathbb{R}^d$;

(2) and environments pick an online convex function $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{x}_t)$, observes some information about f_t and updates the model.

Note that from now on, we use x (and $\mathcal{X}$) instead of w (and $\mathcal{W}$) for consistent to opt. language.

Different Setup

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t$ from a convex set $\mathcal{X} \subseteq \mathbb{R}^d$;

(2) and environments pick an online convex function $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{x}_t)$, observes some information about f_t and updates the model.

on the feedback information:

- *full information*: observe entire f_t (or at least gradient $\nabla f_t(\mathbf{w}_t)$)

- *partial information (bandits)*: observe function value $f_t(\mathbf{w}_t)$ only

less information

full information

horse racing

partial information

multi-armed bandits

Different Setup

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t$ from a convex set $\mathcal{X} \subseteq \mathbb{R}^d$;

(2) and environments pick an online convex function $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{x}_t)$, observes some information about f_t and updates the model.

on the difficulty of environments:

- stochastic setting
- adversarial setting { oblivious / adaptive (non-oblivious)

less restricted but harder

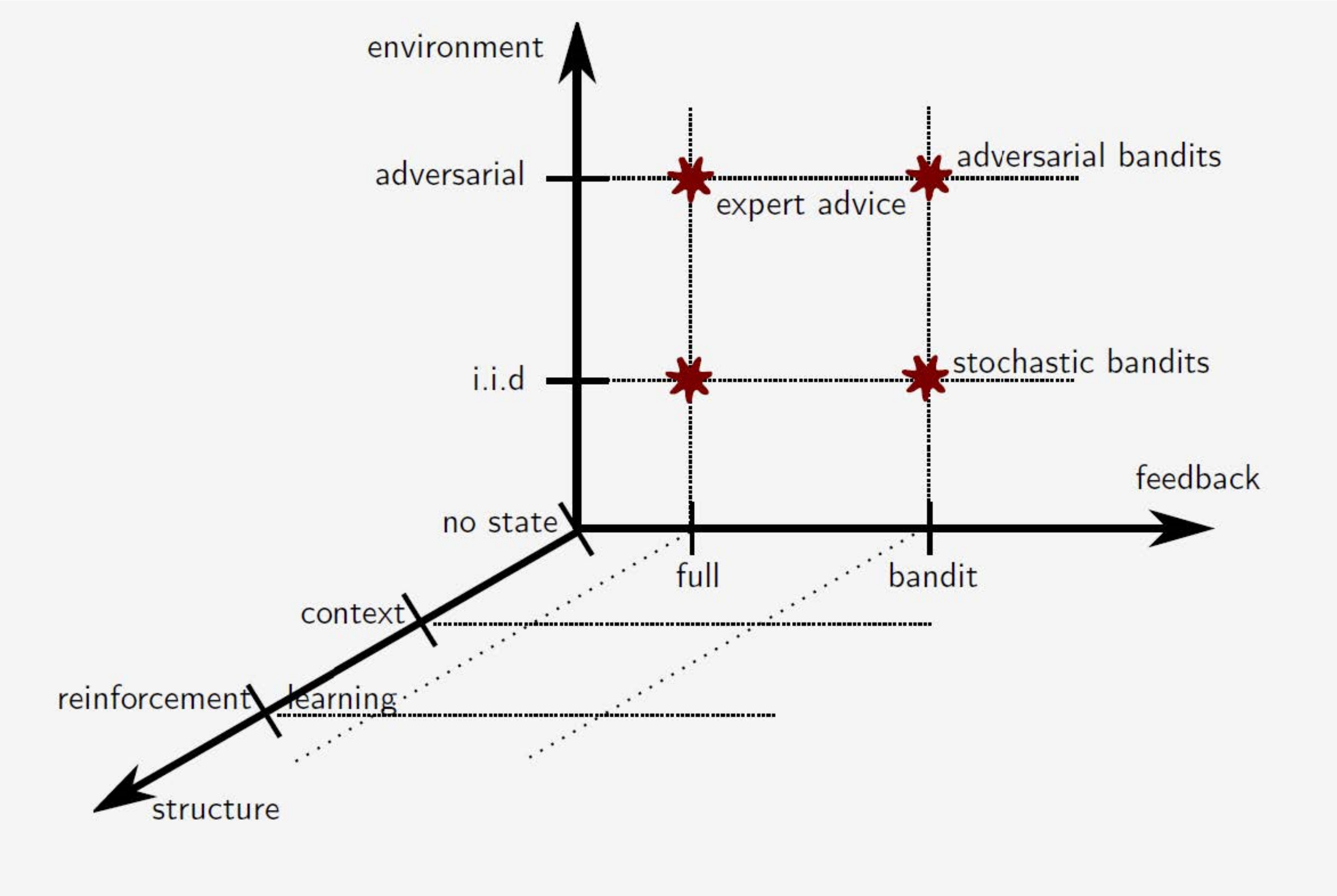
oblivious adversary

examination

adaptive adversary

interview

The Space of Online Learning Problems



Yevgeny Seldin. The Space of Online Learning Problems, ECML-PKDD, Porto, Portugal, 2015.

Online Learning

- Full-information setting:
  - Online Convex Optimization
  - Prediction with Expert Advice
  - ...

- Partial-information setting:
  - Multi-Armed Bandits
  - Linear Bandits/Parametric Bandits
  - Bandit Convex Optimization
  - ...

Online Learning

- **Full-information setting:**
  - Online Convex Optimization
  - Prediction with Expert Advice
  - ...

- Partial-information setting:
  - Multi-Armed Bandits
  - Linear Bandits/Parametric Bandits
  - Bandit Convex Optimization
  - ...

History: Two-Player Zero-Sum Games

Theory of repeated games

James Hannan
(1922–2010)

David Blackwell
(1919–2010)

Learning to play a game (1956)

Play a game repeatedly against a possibly suboptimal opponent

N. Cesa-Bianchi (UNIMI) Online Learning 9 / 49

Zero-sum 2-person games played more than once

	1	2	...	M
1	$\ell(1,1)$	$\ell(1,2)$	...	
2	$\ell(2,1)$	$\ell(2,2)$	...	
⋮	⋮	⋮	⋱	
N				

$N \times M$ known loss matrix

- Row player (player) has N actions
- Column player (opponent) has M actions

For each game round $t = 1, 2, \ldots$

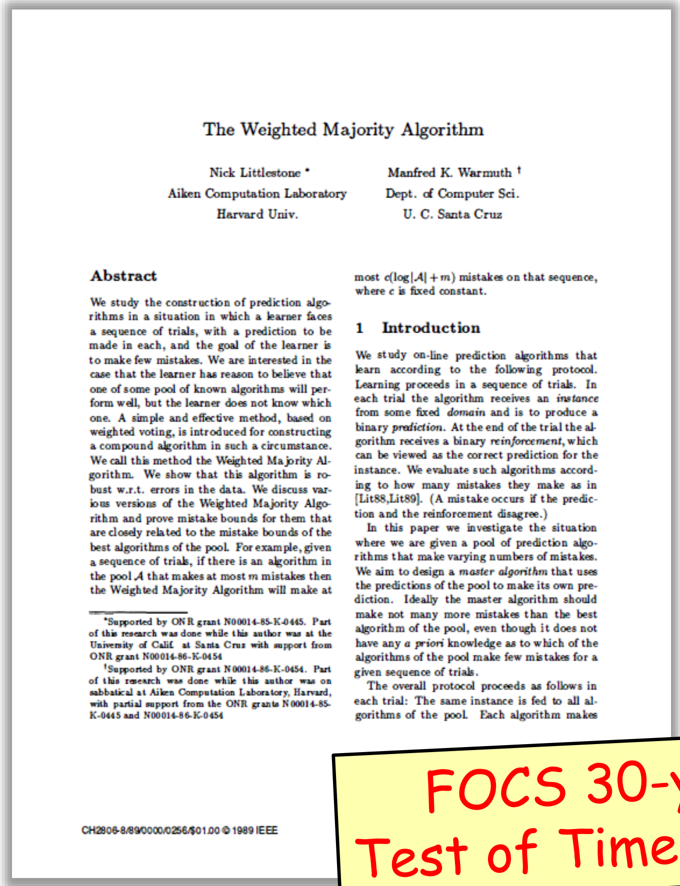
- Player chooses action i_t and opponent chooses action y_t
- The player suffers loss $\ell(i_t, y_t)$ (= gain of opponent)

Player can learn from opponent's history of past choices $y_1, \ldots, y_{t-1}$

N. Cesa-Bianchi (UNIMI) Online Learning 10 / 49

Nicolo Cesa-Bianchi, Online Learning and Online Convex Optimization. Tutorial at the Simons Institute. 2017.

History: Prediction with Expert Advice



The Weighted Majority Algorithm

Nick Littlestone [*]
Aiken Computation Laboratory
Harvard Univ.

Manfred K. Warmuth [†]
Dept. of Computer Sci.
U. C. Santa Cruz

Abstract

We study the construction of prediction algorithms in a situation in which a learner faces a sequence of trials, with a prediction to be made in each, and the goal of the learner is to make few mistakes. We are interested in the case that the learner has reason to believe that one of some pool of known algorithms will perform well, but the learner does not know which one. A simple and effective method, based on weighted voting, is introduced for constructing a compound algorithm in such a circumstance. We call this method the Weighted Majority Algorithm. We show that this algorithm is robust w.r.t. errors in the data. We discuss various versions of the Weighted Majority Algorithm and prove mistake bounds for them that are closely related to the mistake bounds of the best algorithms of the pool. For example, given a sequence of trials, if there is an algorithm in the pool $\mathcal{A}$ that makes at most m mistakes then the Weighted Majority Algorithm will make at most $c(\log|\mathcal{A}| + m)$ mistakes on that sequence, where c is fixed constant.

1 Introduction

We study on-line prediction algorithms that learn according to the following protocol. Learning proceeds in a sequence of trials. In each trial the algorithm receives an *instance* from some fixed *domain* and is to produce a binary *prediction*. At the end of the trial the algorithm receives a binary *reinforcement*, which can be viewed as the correct prediction for the instance. We evaluate such algorithms according to how many mistakes they make as in [Lit88,Lit89]. (A mistake occurs if the prediction and the reinforcement disagree.)

In this paper we investigate the situation where we are given a pool of prediction algorithms that make varying numbers of mistakes. We aim to design a *master algorithm* that uses the predictions of the pool to make its own prediction. Ideally the master algorithm should make not many more mistakes than the best algorithm of the pool, even though it does not have any *a priori* knowledge as to which of the algorithms of the pool make few mistakes for a given sequence of trials.

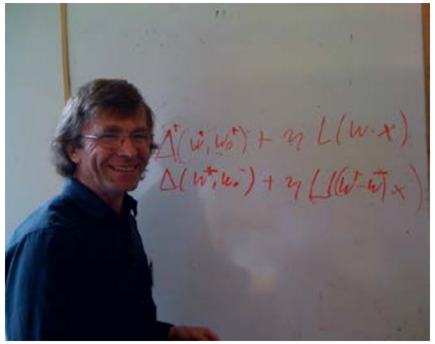
The overall protocol proceeds as follows in each trial: The same instance is fed to all algorithms of the pool. Each algorithm makes

[*] Supported by ONR grant N00014-85-K-0445. Part of this research was done while this author was at the University of Calif. at Santa Cruz with support from ONR grant N00014-86-K-0454

[†] Supported by ONR grant N00014-86-K-0454. Part of this research was done while this author was on sabbatical at Aiken Computation Laboratory, Harvard, with partial support from the ONR grants N00014-85-K-0445 and N00014-86-K-0454

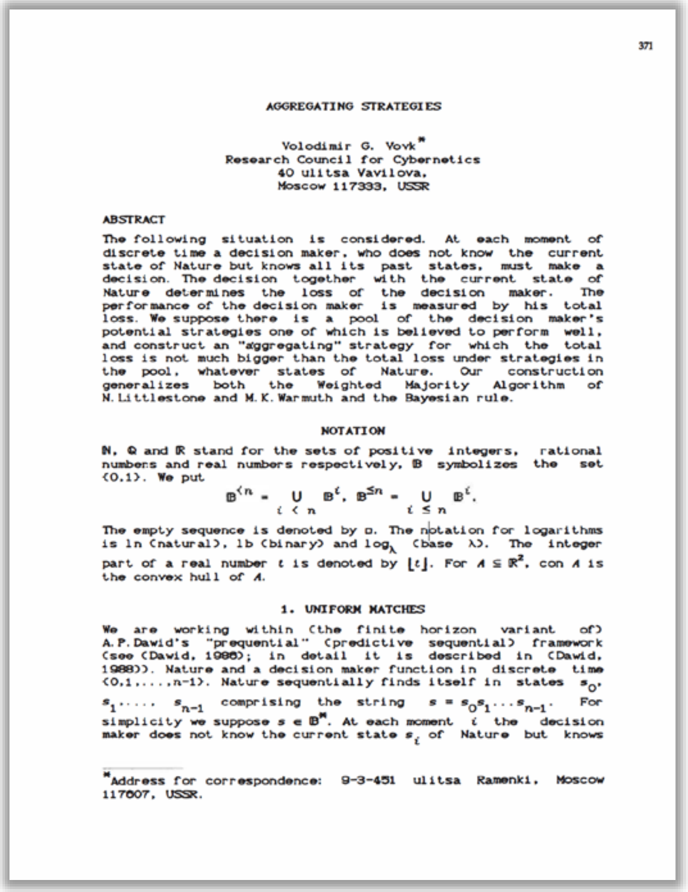
CH2806-8/89/0000/0256$01.00 © 1989 IEEE

FOCS 30-year
Test of Time Award!



Manfred Warmuth

UC Santa Cruz



371

AGGREGATING STRATEGIES

Volodimir G. Vovk [*]
Research Council for Cybernetics
40 ulitsa Vavilova,
Moscow 117333, USSR

ABSTRACT

The following situation is considered. At each moment of discrete time a decision maker, who does not know the current state of Nature but knows all its past states, must make a decision. The decision together with the current state of Nature determines the loss of the decision maker. The performance of the decision maker is measured by his total loss. We suppose there is a pool of the decision maker's potential strategies one of which is believed to perform well, and construct an "aggregating" strategy for which the total loss is not much bigger than the total loss under strategies in the pool, whatever states of Nature. Our construction generalizes both the Weighted Majority Algorithm of N. Littlestone and M. K. Warmuth and the Bayesian rule.

NOTATION

$\mathbb{N}$, $\mathbb{Q}$ and $\mathbb{R}$ stand for the sets of positive integers, rational numbers and real numbers respectively, $\mathbb{B}$ symbolizes the set {0,1}. We put

$$\mathbb{B}^{<n} = \bigcup_{i<n} \mathbb{B}^i, \quad \mathbb{B}^{\le n} = \bigcup_{i \le n} \mathbb{B}^i.$$

The empty sequence is denoted by □. The notation for logarithms is ln (natural), lb (binary) and $\log_\lambda$ (base λ). The integer part of a real number t is denoted by $\lfloor t \rfloor$. For $A \subseteq \mathbb{R}^2$, con A is the convex hull of A.

1. UNIFORM MATCHES

We are working within (the finite horizon variant of) A. P. Dawid's "prequential" (predictive sequential) framework (see (Dawid, 1986); in detail it is described in (Dawid, 1988)). Nature and a decision maker function in discrete time {0,1,...,n-1}. Nature sequentially finds itself in states s_0, s_1, ..., s_{n-1} comprising the string $s = s_0 s_1 \ldots s_{n-1}$. For simplicity we suppose $s \in \mathbb{B}^n$. At each moment i the decision maker does not know the current state s_i of Nature but knows

[*] Address for correspondence: 9-3-451 ulitsa Ramenki, Moscow 117607, USSR.

Volodimir G. Vovk
Royal Holloway,
University of London

Nick Littlestone and Manfred K. Warmuth. "The Weighted Majority Algorithm." FOCS 1989: 256-261.

Volodimir G. Vovk. "Aggregating Strategies." COLT 1990: 371-383.

Online Convex Optimization

- Convex Functions

- Strongly Convex Functions

- Exponentially Concave Functions

Online Convex Optimization

- Convex Functions

- Strongly Convex Functions

- Exponentially Concave Functions

Online Optimization with Convex Functions

Definition 2 (Convex Function). A function $f : \mathcal{X} \mapsto \mathbb{R}$ is convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$

$$\forall \alpha \in [0,1], f((1-\alpha)\mathbf{x} + \alpha \mathbf{y}) \leq (1-\alpha) f(\mathbf{x}) + \alpha f(\mathbf{y}).$$

Equivalently, if f is differentiable, we have that $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

Online Gradient Descent

Online Gradient Descent (OGD)

At each round $t = 1, 2, \cdots$

1. the player first picks a model $\mathbf{x}_t \in \mathcal{X}$;
2. and simultaneously environments pick a ***convex loss function*** $f_t : \mathcal{X} \to \mathbb{R}$;
3. the player suffers loss $f_t(\mathbf{x}_t)$, observes the information (loss) f_t and update the model according to $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)]$.

- $\Pi_{\mathcal{X}}[\mathbf{y}] = \arg\min_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x} - \mathbf{y}\|_2$ denotes the Euclidean projection onto the feasible set $\mathcal{X}$.
- This belongs to the full-information setting, so player can access the gradient $\nabla f_t(\mathbf{x}_t)$.
  But actually the gradient is the only required, so it's also called ***gradient-feedback*** OCO model.

OGD: Regret Analysis

- The following assumptions are required for standard analysis.

Assumption 1 (Convexity). The feasible set $\mathcal{X}$ is closed and convex in Euclidean space, and $f_1, \ldots, f_T$ are convex functions.

Assumption 2 (Bounded Decision Set). The diameter of the set $\mathcal{X}$ is upper bounded by D, i.e., $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}, \|\mathbf{x} - \mathbf{y}\| \leq D$.

Assumption 3 (Bounded Gradient). The norm of the subgradients of f is upper bounded by G, i.e., $\|\nabla f(\mathbf{x})\| \leq G$ for all $\mathbf{x} \in \mathcal{X}$.

OGD: Regret Analysis

Theorem 3 (Regret bound for OGD). *Under Assumptions 1, 2 and 3, online gradient descent (OGD) with step sizes* $\eta_t = \frac{D}{G\sqrt{t}}$ *for* $t \in [T]$ *guarantees:*

$$\text{Regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \le \frac{3}{2} GD\sqrt{T}.$$

The First Gradient Descent Lemma

Lemma 1. *Suppose that f is proper, closed and convex; the feasible domain $\mathcal{X}$ is nonempty, closed and convex. Let $\{\mathbf{x}_t\}_{t=1}^T$ be the sequence generated by the gradient descent method. Then for any $\mathbf{u} \in \mathcal{X}^\star$ and $t \geq 0$,*

$$\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 \leq \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2.$$

Proof:

$$\begin{aligned}
\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 &= \|\Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)] - \mathbf{u}\|^2 \quad \text{(GD)} \\
&\leq \|\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t) - \mathbf{u}\|^2 \quad \text{(Pythagoras Theorem)} \\
&= \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u} \rangle + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2 \\
&\leq \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2
\end{aligned}$$

(convexity: $f_t(\mathbf{x}_t) - f_t(\mathbf{u}) = f_t(\mathbf{x}_t) - f_t(\mathbf{u}) \leq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u} \rangle$) $\square$

Proof for OGD Regret Bound

Proof: We use the first gradient descent lemma to analyze online gradient descent.

Lemma 1. *Suppose that f is proper, closed and convex; the feasible domain $\mathcal{X}$ is nonempty, closed and convex. Let $\{\mathbf{x}_t\}_{t=1}^T$ be the sequence generated by the gradient descent method. Then for any $\mathbf{u} \in \mathcal{X}^\star$ and $t \geq 0$,*

$$\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 \leq \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) + \eta_t^2\|\nabla f_t(\mathbf{x}_t)\|^2.$$

By Lemma 1,

$$2(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) \leq \frac{\|\mathbf{x}_t - \mathbf{u}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{u}\|^2}{\eta_t} + \eta_t G^2$$

Proof for OGD Regret Bound

Proof: By setting $\eta_t = \frac{D}{G\sqrt{t}}$ $\left(\text{ with } \frac{1}{\eta_0} := 0\right)$, summing over T:

$$
\begin{aligned}
2\left(\sum_{t=1}^{T} f_t(\mathbf{x}_t) - f_t(\mathbf{u})\right) &\leq \sum_{t=1}^{T} \frac{\|\mathbf{x}_t - \mathbf{u}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{u}\|^2}{\eta_t} + G^2 \sum_{t=1}^{T} \eta_t & \text{(GD lemma)} \\
&\leq \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{u}\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^{T} \eta_t & (\|\mathbf{x}_{T+1} - \mathbf{u}\|^2 \geq 0) \\
&\leq D^2 \sum_{t=1}^{T} \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) + G^2 \sum_{t=1}^{T} \eta_t \\
&\leq D^2 \frac{1}{\eta_T} + G^2 \sum_{t=1}^{T} \eta_t & \left(\eta_t = \frac{D}{G\sqrt{t}} \text{ and } \textstyle\sum_{t=1}^{T} \frac{1}{\sqrt{t}} \leq 2\sqrt{T}\right) \\
&\leq 3DG\sqrt{T}. & \square
\end{aligned}
$$

Online Convex Optimization

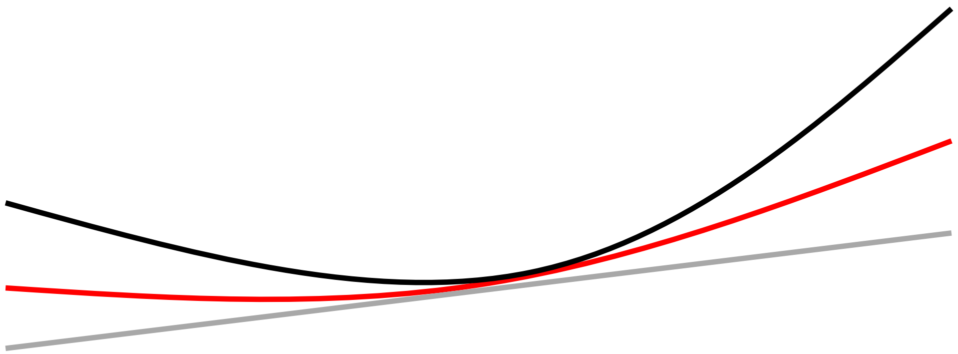
- Convex Functions
- Strongly Convex Functions
- Exponentially Concave Functions

Online Optimization with Strongly Convex Functions

Definition 3 (Strong Convexity). A function f is σ-strongly convex if, for any $\mathbf{x}, \mathbf{y} \in \operatorname{dom} f$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\sigma}{2}\|\mathbf{y} - \mathbf{x}\|^2,$$

or equivalently, $\nabla^2 f(\mathbf{x}) \succeq \alpha I$.



OGD for Strongly Convex Loss

Online Gradient Descent (OGD)

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t \in \mathcal{X}$;

(2) and simultaneously environments pick a ***strongly convex loss*** $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{x}_t)$, observes the information (loss) f_t and update the model according to $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)]$.

- The learning rate for strongly convex OGD is set as $\eta_t = \frac{1}{\sigma t}$.

OGD for Strongly Convex Loss

Theorem 4 (Regret bound for strongly-convex functions). *Under Assumption 1 and Assumption 3, for σ-strongly convex loss functions, online gradient descent with step sizes $\eta_t = \frac{1}{\sigma t}$ achieves the following guarantee*

$$\text{Regret}_T \leq \frac{G^2}{2\sigma}(1 + \log T).$$

- Strongly convex case compared with convex case: $\mathcal{O}(\log T)$ vs. $\mathcal{O}(\sqrt{T})$
- A caveat is that we now don't need Assumption 2 (bounded domain).

OCO with Strongly Convex Functions

Proof: we start by extending *the first GD lemma* to strongly convex case.

Strongly convex case:

$$\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 \leq \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u} \rangle + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2$$

$$\leq \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t \left(f_t(\mathbf{x}_t) - f_t(\mathbf{u}) + \frac{\sigma}{2}\|\mathbf{x}_t - \mathbf{u}\|^2 \right) + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2$$

(strong convexity: $f_t(\mathbf{x}_t) - f_t(\mathbf{u}) + \frac{\sigma}{2}\|\mathbf{x}_t - \mathbf{u}\|^2 \leq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{u} \rangle$)

$$\leq (1 - \sigma\eta_t) \|\mathbf{x}_t - \mathbf{u}\|^2 - 2\eta_t (f_t(\mathbf{x}_t) - f_t(\mathbf{u})) + \eta_t^2 \|\nabla f_t(\mathbf{x}_t)\|^2$$

$$\Longrightarrow f_t(\mathbf{x}_t) - f_t(\mathbf{u}) \leq \frac{\eta_t^{-1} - \sigma}{2}\|\mathbf{x}_t - \mathbf{u}\|^2 - \frac{\eta_t^{-1}}{2}\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 + \frac{\eta_t G^2}{2}$$

(rearranging)

OCO with Strongly Convex Functions

Proof: $f_t(\mathbf{x}_t) - f_t(\mathbf{u}) \leq \frac{\eta_t^{-1} - \sigma}{2}\|\mathbf{x}_t - \mathbf{u}\|^2 - \frac{\eta_t^{-1}}{2}\|\mathbf{x}_{t+1} - \mathbf{u}\|^2 + \frac{\eta_t G^2}{2}$

Summing from $t = 1$ to T, setting $\eta_t = \frac{1}{\sigma t}$ (define $\frac{1}{\eta_0} := 0\Big)$:

$$
\begin{aligned}
2\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{u})\right) &\leq \sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{u}\|^2\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \sigma\right) + G^2\sum_{t=1}^{T}\eta_t \quad \left(\tfrac{1}{\eta_0} := 0\right) \\
&= 0 + G^2\sum_{t=1}^{T}\frac{1}{\sigma t} \quad \left(\tfrac{1}{\eta_0} \triangleq 0, \|\mathbf{x}_{T+1} - \mathbf{u}\|^2 \geq 0, \tfrac{1}{\eta_t} - \tfrac{1}{\eta_{t-1}} - \sigma = 0\right) \\
&\leq \frac{G^2}{\sigma}(1 + \log T). \qquad \square
\end{aligned}
$$

Online Convex Optimization

- Convex Functions

- Strongly Convex Functions

- Exponentially Concave Functions

Convergence of Proximal Gradient

Convex Problem

Property: $f_t(\mathbf{x}) \geq f_t(\mathbf{y}) + \nabla f_t(\mathbf{y})^\top(\mathbf{x}-\mathbf{y})$

OGD: $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \frac{1}{\sqrt{t}}\nabla f_t(\mathbf{x}_t)\right]$

$$\text{Regret}_T \leq \frac{3}{2}GD\sqrt{T}$$

Strongly Convex Problem

Property: $f_t(\mathbf{y}) \geq f_t(\mathbf{x}) + \nabla f_t(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) + \frac{\sigma}{2}\|\mathbf{y}-\mathbf{x}\|^2$

OGD: $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \frac{1}{\sigma t}\nabla f_t(\mathbf{x}_t)\right]$

$$\text{Regret}_T \leq \frac{G^2}{2\sigma}(1+\log T)$$

Can we explore more function class with a regret rate faster than $\sqrt{T}$?

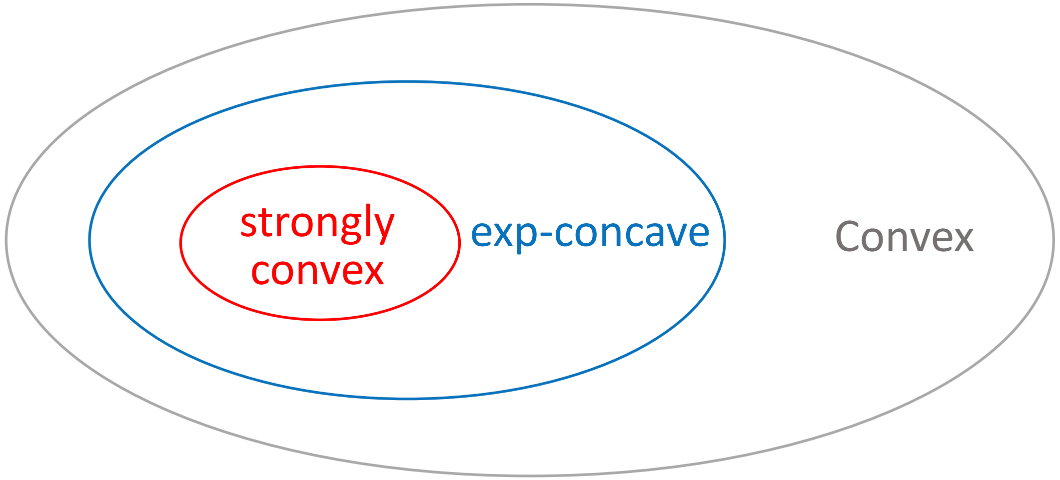
Exponentially-concave Function

Definition 2 (Exp-concavity). A convex function $f : \mathbb{R}^d \mapsto \mathbb{R}$ is defined to be α-exp-concave over $\mathcal{X} \subseteq \mathbb{R}^d$ if the function g is concave, where $g : \mathcal{X} \mapsto \mathbb{R}$ is defined as

$$g(\mathbf{x}) = e^{-\alpha f(\mathbf{x})}.$$

Directly employ OGD: Regret bound $\mathcal{O}(\sqrt{T})$.

But actually we can get a **tighter** bound!



An Example for Exp-concave Learning

- Universal Portfolio Selection 
  - a total of d stocks in the stock market.
  - each round, the player chooses stocks by a distribution $\mathbf{x}_t \in \Delta_d$.
  - the market returns the price ratio $\boldsymbol{\theta}_t$ between iter t and $t+1$,

$$\boldsymbol{\theta}_t(i) = \frac{\text{price of stock}_i \text{ at time } t+1}{\text{price of stock}_i \text{ at time } t}$$

which means that our final wealth W_T will be: $W_T = W_1 \cdot \prod_{t=1}^{T} \boldsymbol{\theta}_t^\top \mathbf{x}_t$

$\Rightarrow$ Our goal is to maximize our wealth at time T.

An Example for Exp-concave Learning

- Universal Portfolio Selection 

  - we hope to maximize the logarithm of W_T

$$\log \frac{W_T}{W_1} = \sum_{t=1}^{T} \log \boldsymbol{\theta}_t^\top \mathbf{x}_t$$

  - using OCO framework,

$$f_t(\mathbf{x}) = \log(\boldsymbol{\theta}_t^\top \mathbf{x})$$

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t \in \Delta_d$;

(2) and simultaneously environments pick an online function $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player get a ***gain*** $f_t(\mathbf{x}_t) = \log(\boldsymbol{\theta}_t^\top \mathbf{x}_t)$, observes f_t and updates the model.

- Goal: $\text{Regret}_T = \max_{\mathbf{x}^\star \in \Delta_d} \sum_{t=1}^{T} f_t(\mathbf{x}^\star) - \sum_{t=1}^{T} f_t(\mathbf{x}_t)$

online function is exp-concave

Exponential-concave Function

Lemma 3 (Property of Exp-concavity). *Let $f : \mathcal{X} \to \mathbb{R}$ be an α-exp-concave function, and D, G denote the diameter of $\mathcal{X}$ and a bound on the (sub)gradients of f respectively. The following holds for all $\gamma \leq \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ and all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:*

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}(\mathbf{x}-\mathbf{y})^\top \nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top (\mathbf{x}-\mathbf{y}).$$

Proof. Recall that f is α-exp-concave if and only if $e^{-\alpha f(\mathbf{x})}$ is concave.

As $2\gamma \leq \alpha$, $e^{-2\gamma f(\mathbf{x})} = \left(e^{-\alpha f(\mathbf{x})}\right)^{2\gamma/\alpha}$ is also concave and thus is 2γ-exp-concave.

$$e^{-2\gamma f(\mathbf{x})} - e^{-2\gamma f(\mathbf{y})} \leq \left\langle \mathbf{x}-\mathbf{y}, -2\gamma e^{-2\gamma f(\mathbf{y})} \nabla f(\mathbf{y}) \right\rangle .$$

(concavity)

Exponential-concave Function

Lemma 3 (Property of Exp-concavity). *Let $f : \mathcal{X} \to \mathbb{R}$ be an α-exp-concave function, and D, G denote the diameter of $\mathcal{X}$ and a bound on the (sub)gradients of f respectively. The following holds for all $\gamma \leq \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ and all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:*

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}(\mathbf{x}-\mathbf{y})^\top \nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top (\mathbf{x}-\mathbf{y}).$$

Proof. Dividing $e^{-2\gamma f(\mathbf{y})}$ at both sides achieves

$$f(\mathbf{y}) - f(\mathbf{x}) \leq \frac{1}{2\gamma}\log\left(1 + 2\gamma\langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{y})\rangle\right).$$

Our constructive condition $\gamma \leq \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ ensures $|2\gamma\langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{y})\rangle| \leq 1$,

$$f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{y})\rangle - \frac{\gamma}{2}\langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{y})\rangle^2$$

($\log(1+x) \leq x - \frac{1}{4}x^2$) holds for ($|x| \leq 1$) □

A Comparison of Different Curvatures

- Convex

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y})$$

- Strongly Convex

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\sigma}{2} \|\mathbf{x} - \mathbf{y}\|_2^2$$

- Exponentially Concave

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\gamma}{2} (\mathbf{x} - \mathbf{y})^\top \nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) \\ &= f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\gamma}{2} \|\mathbf{x} - \mathbf{y}\|^2_{\nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top} \end{aligned}$$

Exponential-concave Function

Lemma 3 (Property of Exp-concavity). *Let $f : \mathcal{X} \to \mathbb{R}$ be an α-exp-concave function, and D, G denote the diameter of $\mathcal{X}$ and a bound on the (sub)gradients of f respectively. The following holds for all $\gamma \leq \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ and all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$:*

$$
\begin{aligned}
f(\mathbf{x}) &\geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}(\mathbf{x}-\mathbf{y})^\top \nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top (\mathbf{x}-\mathbf{y}) \\
&= f(\mathbf{y}) + \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}\|\mathbf{x}-\mathbf{y}\|^2_{\nabla f(\mathbf{y}) \nabla f(\mathbf{y})^\top}
\end{aligned}
$$

Algorithmic intuition:

- For convex loss, we use 2-norm to encode the structure of the space.
- Can we exploit ***local structures*** of exp-concave loss to improve the regret?

ONS for Exp-concave Function

Online Newton Step

Input: parameters $\gamma, \varepsilon > 0$, matrix $A_0 = \varepsilon I_d$

At each round $t = 1, 2, \cdots$

(1) the player first picks a model $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$;

(2) and simultaneously environments pick an *exp-concave loss function* $f_t : \mathcal{X} \to \mathbb{R}$;

(3) the player suffers loss $f_t(\mathbf{x}_t)$, observes the information (loss) f_t and update:

$$\text{Update } A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t) \nabla f_t(\mathbf{x}_t)^\top$$

$$\text{Update } \mathbf{x}_{t+1} = \arg\min_{\mathbf{x} \in \mathcal{X}} \left\| \mathbf{x} - \left(\mathbf{x}_t - \frac{1}{\gamma} A_t^{-1} \nabla f_t(\mathbf{x}_t) \right) \right\|_{A_t}^2$$

In a View of Proximal Gradient

Convex Problem

Property: $f_t(\mathbf{x}) \geq f_t(\mathbf{y}) + \nabla f_t(\mathbf{y})^\top(\mathbf{x}-\mathbf{y})$

OGD: $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \frac{1}{\sqrt{t}}\nabla f_t(\mathbf{x}_t)\right]$

Proximal type update:

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}, \nabla f_t(\mathbf{x}_t)\rangle + \frac{1}{2\eta_t}\|\mathbf{x}-\mathbf{x}_t\|_2^2$$

Exp-concave Problem

Property: $f_t(\mathbf{x}) \geq f_t(\mathbf{y}) + \nabla f_t(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}\|\mathbf{x}-\mathbf{y}\|^2_{\nabla f_t(\mathbf{y})\nabla f_t(\mathbf{y})^\top}$

ONS: $A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$

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

Proximal type update:

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}, \nabla f_t(\mathbf{x}_t)\rangle + \frac{\gamma}{2}\|\mathbf{x}-\mathbf{x}_t\|_{A_t}^2$$

In a View of Proximal Gradient

Proof.

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}^{A_t}\left[\mathbf{x}_t - \frac{A_t^{-1}}{\gamma}\mathbf{g}_t\right] \quad (\mathbf{g}_t = \nabla f_t(\mathbf{x}_t))$$

$$= \arg\min_{\mathbf{x}\in\mathcal{X}} \left(\mathbf{x} - \mathbf{x}_t + \frac{A_t^{-1}}{\gamma}\mathbf{g}_t\right)^\top A_t \left(\mathbf{x} - \mathbf{x}_t + \frac{A_t^{-1}}{\gamma}\mathbf{g}_t\right)$$

$$= \arg\min_{\mathbf{x}\in\mathcal{X}} \left(\mathbf{x} - \mathbf{x}_t + \frac{A_t^{-1}}{\gamma}\mathbf{g}_t\right)^\top \left(A_t\mathbf{x} - A_t\mathbf{x}_t + \frac{\mathbf{g}_t}{\gamma}\right)$$

$$= \arg\min_{\mathbf{x}\in\mathcal{X}} (\mathbf{x} - \mathbf{x}_t)^\top A_t (\mathbf{x} - \mathbf{x}_t) + \cancel{(A^{-1})^\top \mathbf{g}_t^\top \mathbf{g}_t} + 2\frac{\mathbf{g}_t^\top(\mathbf{x} - \cancel{\mathbf{x}_t})}{\gamma}$$

(constant)

$$= \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}_t, \mathbf{g}_t\rangle + \frac{\gamma}{2}\|\mathbf{x} - \mathbf{x}_t\|_{A_t}^2$$

Exp-concave Problem

Property: $f_t(\mathbf{x}) \geq f_t(\mathbf{y}) + \nabla f_t(\mathbf{y})^\top(\mathbf{x} - \mathbf{y}) + \frac{\gamma}{2}\|\mathbf{x} - \mathbf{y}\|^2_{\nabla f_t(\mathbf{y})\nabla f_t(\mathbf{y})^\top}$

ONS: $A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$

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

Proximal type update:

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}, \nabla f_t(\mathbf{x}_t)\rangle + \frac{\gamma}{2}\|\mathbf{x} - \mathbf{x}_t\|_{A_t}^2$$

ONS for Exp-concave Function

Theorem 5. *Under Assumptions 1, 2 and 3, for α-exp-concave online functions, the ONS algorithm with parameters $\gamma = \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ and $\varepsilon = \frac{1}{\gamma^2 D^2}$ (recall that the initial matrix is $A_0 = \varepsilon I_d$) guarantees*

$$\text{Regret}_T \leq \mathcal{O}\left((\frac{1}{\alpha} + GD)d\log T\right),$$

where d is the dimension of the feasible domain $\mathcal{X} \subseteq \mathbb{R}^d$.

OCO with Exp-concave Functions

- $A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$
- $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \left\|\mathbf{x} - (\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\mathbf{g}_t)\right\|_{A_t}^2$

Extending *the first GD lemma* to ***exp-concave case***:

Proof. We use norm induced by A_t instead of 2-norm.

$$\|\mathbf{x}_{t+1} - \mathbf{u}\|_{A_t}^2 = \left\|\Pi_{\mathcal{X}}^{A_t}\left[\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t)\right] - \mathbf{u}\right\|_{A_t}^2 \quad (\Pi_{\mathcal{X}}^{A}[\mathbf{y}] \triangleq \arg\min_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}-\mathbf{y}\|_A^2)$$

$$\leq \left\|\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t) - \mathbf{u}\right\|_{A_t}^2 \quad \text{(A_t is semidefinite matrix) (Pythagoras theorem)}$$

$$= \left(\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t) - \mathbf{u}\right)^\top A_t \left(\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t) - \mathbf{u}\right) \quad \text{(definition of $\|\cdot\|_{A_t}^2$)}$$

$$= \left(\mathbf{x}_t - \mathbf{u} - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t)\right)^\top \left(A_t(\mathbf{x}_t - \mathbf{u}) - \frac{1}{\gamma}\nabla f_t(\mathbf{x}_t)\right)$$

OCO with Exp-concave Functions

- $A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$
- $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \left\| \mathbf{x} - (\mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\mathbf{g}_t) \right\|_{A_t}^2$

Extending *the first GD lemma* to *exp-concave case*:

Proof.

$$\|\mathbf{x}_{t+1} - \mathbf{u}\|_{A_t}^2 = \left(\mathbf{x}_t - \mathbf{u} - \frac{1}{\gamma}A_t^{-1}\nabla f_t(\mathbf{x}_t)\right)^\top \left(A_t(\mathbf{x}_t - \mathbf{u}) - \frac{1}{\gamma}\nabla f_t(\mathbf{x}_t)\right)$$

$$= (\mathbf{x}_t - \mathbf{u})^\top A_t (\mathbf{x}_t - \mathbf{u}) - \frac{2}{\gamma}\nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{u}) + \frac{1}{\gamma^2}\nabla f_t(\mathbf{x}_t)^\top A_t^{-1}\nabla f_t(\mathbf{x}_t)$$

$$\leq \|\mathbf{x}_t - \mathbf{u}\|_{A_t}^2 - \frac{2}{\gamma}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{u})\right) + \frac{1}{\gamma^2}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$$

$$- (\mathbf{x}_t - \mathbf{u})^\top \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}_t - \mathbf{u})$$

(Exp-concave: $f_t(\mathbf{x}) \geq f_t(\mathbf{y}) + \nabla f_t(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\gamma}{2}(\mathbf{x}-\mathbf{y})^\top \nabla f_t(\mathbf{y})\nabla f_t(\mathbf{y})^\top(\mathbf{x}-\mathbf{y})$)

OCO with Exp-concave Functions

Proof. $$\|\mathbf{x}_{t+1} - \mathbf{u}\|_{A_t}^2$$

$$\leq \|\mathbf{x}_t - \mathbf{u}\|_{A_t}^2 - \frac{2}{\gamma}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{u})\right) - \|\mathbf{x}_t - \mathbf{u}\|^2_{\nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top} + \frac{1}{\gamma^2}\|\nabla f_t(\mathbf{x}_t)\|^2_{A_t^{-1}}$$

$$\Rightarrow f_t(\mathbf{x}_t) - f_t(\mathbf{u}) \leq \frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{u}\|_{A_t}^2 - \frac{\gamma}{2}\|\mathbf{x}_{t+1} - \mathbf{u}\|_{A_t}^2 - \frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{u}\|^2_{\nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top} + \frac{1}{2\gamma}\|\nabla f_t(\mathbf{x}_t)\|^2_{A_t^{-1}}$$

(rearranging)

Summing from $t = 1$ to T, by telescoping:

$$\sum_{t=1}^T (f_t(\mathbf{x}_t) - f_t(\mathbf{u})) \leq \frac{\gamma}{2}\sum_{t=1}^T\left(\|\mathbf{x}_t - \mathbf{u}\|_{A_t}^2 - \|\mathbf{x}_t - \mathbf{u}\|_{A_{t-1}}^2\right) + \frac{1}{2\gamma}\sum_{t=1}^T\|\nabla f_t(\mathbf{x}_t)\|^2_{A_t^{-1}}$$

$$+ \frac{\gamma}{2}\|\mathbf{x}_1 - \mathbf{u}\|_{A_0}^2 - \frac{\gamma}{2}\sum_{t=1}^T\|\mathbf{x}_t - \mathbf{u}\|^2_{\nabla f(\mathbf{x}_t)\nabla f(\mathbf{x}_t)^\top}$$

$$\leq \frac{\gamma}{2}\|\mathbf{x}_1 - \mathbf{u}\|_{A_0}^2 + \frac{1}{2\gamma}\sum_{t=1}^T\|\nabla f_t(\mathbf{x}_t)\|^2_{A_t^{-1}}$$

$(A_t = A_{t-1} + \nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top)$

OCO with Exp-concave Functions

Proof.

$$\sum_{t=1}^{T}(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) \le \frac{\gamma}{2}\|\mathbf{x}_1 - \mathbf{u}\|_{A_0}^2 + \frac{1}{2\gamma}\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$$

By the definition that $A_0 \triangleq \varepsilon I_d$, $\varepsilon = \frac{1}{\gamma^2 D^2}$ and the diameter $\|\mathbf{x}_1 - \mathbf{u}\|_2^2 \le D^2$:

$$\begin{aligned}\sum_{t=1}^{T}(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) &\le \frac{\gamma}{2}(\mathbf{x}_1 - \mathbf{u})^\top A_0 (\mathbf{x}_1 - \mathbf{u}) + \frac{1}{2\gamma}\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2 \\ &\le \frac{1}{2\gamma} + \frac{1}{2\gamma}\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2 .\end{aligned}$$

Next, we bound the term $\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

OCO with Exp-concave Functions

Proof. Next, we bound the term $\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

Lemma 4 **(Elliptical Potential Lemma).** *For any sequence* $\{X_1, \ldots, X_T\} \in \mathbb{R}^{d \times T}$, *suppose* $U_0 = \lambda I, U_t = U_{t-1} + X_t X_t^\top$, *and* $\|X_t\|_2 \leq L$, *then*

$$\sum_{t=1}^{T} \|X_t\|_{U_t^{-1}}^2 \leq d \log \left(1 + \frac{L^2 T}{\lambda d}\right)$$

Proof. $U_{t-1} = U_t - X_t X_t^\top = U_t^{\frac{1}{2}} \left(I - U_t^{-\frac{1}{2}} X_t X_t^\top U_t^{-\frac{1}{2}}\right) U_t^{\frac{1}{2}}$ (definition of U_t)

$$\det(U_{t-1}) = \det(U_t) \det \left(I - U_t^{-\frac{1}{2}} X_t X_t^\top U_t^{-\frac{1}{2}}\right)$$ (determinant on both side)

OCO with Exp-concave Functions

Proof. Next, we bound the term $\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

Lemma 5. *For any* $\mathbf{v} \in \mathbb{R}^d$*, we have*

$$\det\left(I - \mathbf{v}\mathbf{v}^\top\right) = 1 - \|\mathbf{v}\|_2^2$$

Proof.

(i) $\left(I - \mathbf{v}\mathbf{v}^\top\right)\mathbf{v} = \left(1 - \|\mathbf{v}\|_2^2\right)\mathbf{v}$, therefore, $\mathbf{v}$ is its eigenvector with $\left(1 - \|\mathbf{v}\|_2^2\right)$ as eigenvalue;

(ii) $\left(I - \mathbf{v}\mathbf{v}^\top\right)\mathbf{v}^\perp = \mathbf{v}^\perp$, therefore, $\mathbf{v}^\perp \perp \mathbf{v}$ is its eigenvector with 1 as the eigenvalue.

OCO with Exp-concave Functions

Proof. Next, we bound the term $\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

Lemma 4 (Elliptical Potential Lemma). *For any sequence* $\{X_1, \ldots, X_T\} \in \mathbb{R}^{d \times T}$, *suppose* $U_0 = \lambda I, U_t = U_{t-1} + X_t X_t^\top$, *and* $\|X_t\|_2 \leq L$, *then*

$$\sum_{t=1}^{T} \|X_t\|_{U_t^{-1}}^2 \leq d \log \left(1 + \frac{L^2 T}{\lambda d}\right)$$

Proof. $\det(U_{t-1}) = \det(U_t) \det\left(I - U_t^{-\frac{1}{2}} X_t X_t^\top U_t^{-\frac{1}{2}}\right) = \det(U_t)\left(1 - \left\|U_t^{-\frac{1}{2}} X_t\right\|_2^2\right)$

(by Lemma 5)

$$\Rightarrow \|X_t\|_{U_t^{-1}}^2 = \left\|U_t^{-\frac{1}{2}} X_t\right\|_2^2 = 1 - \frac{\det(U_{t-1})}{\det(U_t)}$$

(rearranging, U is a symmetric matrix)

OCO with Exp-concave Functions

Proof. Next, we bound the term $\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

Lemma 4 (Elliptical Potential Lemma). *For any sequence $\{X_1, \ldots, X_T\} \in \mathbb{R}^{d \times T}$, suppose $U_0 = \lambda I, U_t = U_{t-1} + X_t X_t^\top$, and $\|X_t\|_2 \leq L$, then*

$$\sum_{t=1}^{T} \|X_t\|_{U_t^{-1}}^2 \leq d \log\left(1 + \frac{L^2 T}{\lambda d}\right)$$

Proof.

$$\Rightarrow \sum_{t=1}^{T} X_t^\top U_t^{-1} X_t = \sum_{t=1}^{T} \left(1 - \frac{\det(U_{t-1})}{\det(U_t)}\right) \leq \sum_{t=1}^{T} \log \frac{\det(U_t)}{\det(U_{t-1})} \quad (\forall x > 0, 1 - x \leq -\log x)$$

$$= \log \frac{\det(U_T)}{\det(U_0)} = d \log\left(1 + \frac{L^2 T}{\lambda d}\right)$$

$\mathrm{Tr}(U_T) \leq \mathrm{Tr}(U_0) + L^2 T = \lambda d + L^2 T$

$\Rightarrow \det(U_T) \leq (\lambda + L^2 T / d)^d$

OCO with Exp-concave Functions

Proof. Next, we bound the term $\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2$.

Lemma 4 (Elliptical Potential Lemma). *For any sequence* $\{X_1, \ldots, X_T\} \in \mathbb{R}^{d \times T}$, *suppose* $U_0 = \lambda I, U_t = U_{t-1} + X_t X_t^\top$, *and* $\|X_t\|_2 \leq L$, *then*

$$\sum_{t=1}^{T} \|X_t\|_{U_t^{-1}}^2 \leq d \log\left(1 + \frac{L^2 T}{\lambda d}\right)$$

Therefore, by Lemma 4, we have

$$\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2 \leq d \log\left(1 + \frac{D^2 T}{\varepsilon d}\right).$$

OCO with Exp-concave Functions

Proof. To conclude,

$$\sum_{t=1}^{T}(f_t(\mathbf{x}_t) - f_t(\mathbf{u})) \leq \underbrace{\frac{\gamma}{2}\|\mathbf{x}_1 - \mathbf{u}\|_{A_0}^2}_{\leq \frac{1}{2\gamma}} + \underbrace{\frac{1}{2\gamma}\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_{A_t^{-1}}^2}_{\leq \frac{d}{2\gamma}\log\left(1+\frac{D^2T}{\varepsilon d}\right)}.$$

(bounded domain) (elliptical potential lemma)

Recall that $\gamma = \frac{1}{2}\min\left\{\frac{1}{GD}, \alpha\right\}$ and $\varepsilon = \frac{1}{\gamma^2 D^2}$,

$$\text{Regret}_T \leq \mathcal{O}\left((\frac{1}{\alpha} + GD)d\log T\right). \qquad \square$$

Lower Bounds

- A natural question: whether previous regret can be improved?

- Lower bound argument:

  minimax bound: smallest possible worst-case regret of any algorithm:

$$\min_{\mathcal{A}} \max_{\ell_1,\ldots,\ell_T} \mathrm{Regret}_T$$

Theorem 7 (Lower Bound for OCO). *Any algorithm for online convex optimization incurs $\Omega(DG\sqrt{T})$ regret in the worst case. This is true even if the cost functions are generated from a fixed stationary distribution.*

Lower Bounds

Theorem 7 (Lower Bound for OCO). *Any algorithm for online convex optimization incurs $\Omega(DG\sqrt{T})$ regret in the worst case. This is true even if the cost functions are generated from a fixed stationary distribution.*

Proof Sketch.

Construct a **'hard'** environment:

- Binary classification, loss functions in each iteration are chosen at random
- Similar results can be obtained for strongly convex and exp-concave cases

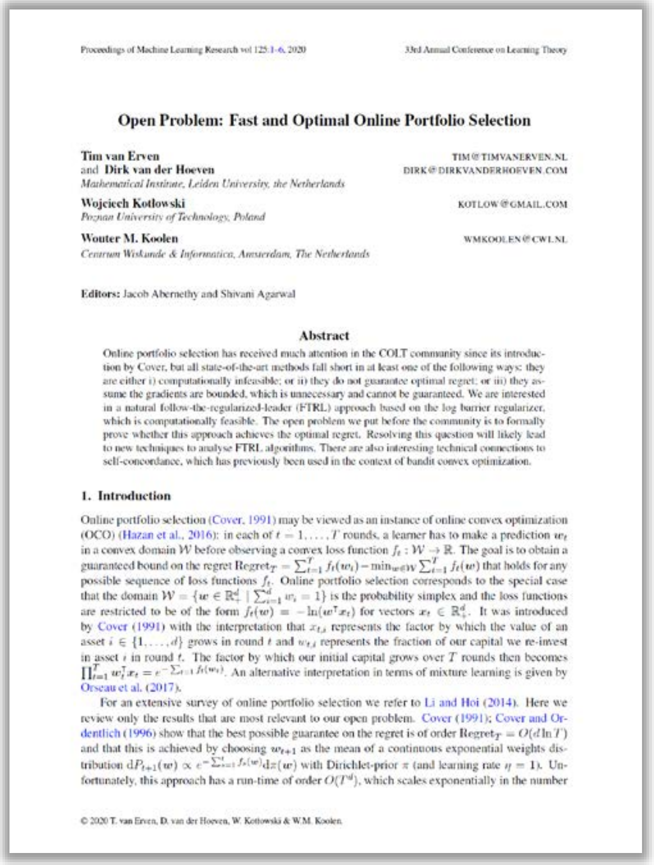
Comparison

	Algorithm	**Upper Bound**	**Lower Bound**
Convex	OGD	$\mathcal{O}(\sqrt{T})$	$\Omega(\sqrt{T})$
σ-Strongly Convex	OGD	$\mathcal{O}(\frac{\log T}{\sigma})$	$\Omega(\frac{\log T}{\sigma})$
α-Exp-concave	ONS	$\mathcal{O}(\frac{d \log T}{\alpha})$	$\Omega(\frac{d \log T}{\alpha})$

Back to Exp-concave Learning

- Universal Portfolio Selection 

Algorithm	Regret	Runtime (per round)
Universal Portfolios	$d \log(T)$	$d^4 T^{14}$
Online Gradient Descent	$G_2 \sqrt{T}$	d
Exponentiated Gradient	$G_\infty \sqrt{T \log(d)}$	d
Online Newton Step (ONS)	$G_\infty d \log(T)$	$d^2 +$ generalized projection on Δ_d
Soft-Bayes	$\sqrt{dT \log(d)}$	d
Ada-BARRONS	$d^2 \log^4(T)$	$d^{2.5} T$
BISONS	$d^2 \log^2(T)$	$\text{poly}(d)$
AdaMix+DONS	$d^2 \log^5(T)$	d^3
VB-FTRL	$d \log(T)$	$d^2 T$



Proceedings of Machine Learning Research vol 125:1–6, 2020 33rd Annual Conference on Learning Theory

Open Problem: Fast and Optimal Online Portfolio Selection

Tim van Erven TIM@TIMVANERVEN.NL
and **Dirk van der Hoeven** DIRK@DIRKVANDERHOEVEN.COM
Mathematical Institute, Leiden University, the Netherlands
Wojciech Kotłowski KOTLOW@GMAIL.COM
Poznan University of Technology, Poland
Wouter M. Koolen WMKOOLEN@CWI.NL
Centrum Wiskunde & Informatica, Amsterdam, The Netherlands

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

Online portfolio selection has received much attention in the COLT community since its introduction by Cover, but all state-of-the-art methods fall short in at least one of the following ways: they are either i) computationally infeasible; or ii) they do not guarantee optimal regret; or iii) they assume the gradients are bounded, which is unnecessary and cannot be guaranteed. We are interested in a natural follow-the-regularized-leader (FTRL) approach based on the log barrier regularizer, which is computationally feasible. The open problem we put before the community is to formally prove whether this approach achieves the optimal regret. Resolving this question will likely lead to new techniques to analyse FTRL algorithms. There are also interesting technical connections to self-concordance, which has previously been used in the context of bandit convex optimization.

1. Introduction

Online portfolio selection (Cover, 1991) may be viewed as an instance of online convex optimization (OCO) (Hazan et al., 2016): in each of $t = 1, \dots, T$ rounds, a learner has to make a prediction w_t in a convex domain $\mathcal{W}$ before observing a convex loss function $f_t : \mathcal{W} \to \mathbb{R}$. The goal is to obtain a guaranteed bound on the regret $\text{Regret}_T = \sum_{t=1}^T f_t(w_t) - \min_{w \in \mathcal{W}} \sum_{t=1}^T f_t(w)$ that holds for any possible sequence of loss functions f_t. Online portfolio selection corresponds to the special case that the domain $\mathcal{W} = \{w \in \mathbb{R}_+^d \mid \sum_{i=1}^d w_i = 1\}$ is the probability simplex and the loss functions are restricted to be of the form $f_t(w) = -\ln(w^\top x_t)$ for vectors $x_t \in \mathbb{R}_+^d$. It was introduced by Cover (1991) with the interpretation that $x_{t,i}$ represents the factor by which the value of an asset $i \in \{1, \dots, d\}$ grows in round t and $w_{t,i}$ represents the fraction of our capital we re-invest in asset i in round t. The factor by which our initial capital grows over T rounds then becomes $\prod_{t=1}^T w_t^\top x_t = e^{-\sum_{t=1}^T f_t(w_t)}$. An alternative interpretation in terms of mixture learning is given by Orseau et al. (2017).

For an extensive survey of online portfolio selection we refer to Li and Hoi (2014). Here we review only the results that are most relevant to our open problem. Cover (1991); Cover and Ordentlich (1996) show that the best possible guarantee on the regret is of order $\text{Regret}_T = O(d \ln T)$ and that this is achieved by choosing w_{t+1} as the mean of a continuous exponential weights distribution $\mathrm{d}P_{t+1}(w) \propto e^{-\sum_{s=1}^t f_s(w)} \mathrm{d}\pi(w)$ with Dirichlet-prior π (and learning rate $\eta = 1$). Unfortunately, this approach has a run-time of order $O(T^d)$, which scales exponentially in the number

© 2020 T. van Erven, D. van der Hoeven, W. Kotłowski & W.M. Koolen.

[COLT 2020 Open Problem]

$\Rightarrow$ still an important open problem: efficiency and optimality

Application to Stochastic Optimization

- Consider the following ***convex optimization*** problem:

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$$

- Stochastic optimization method

Computational oracle: only access ***noisy*** gradient oracle, namely, $\mathbf{g}(\mathbf{x})$, such that

$$\mathbb{E}[\mathbf{g}(\mathbf{x})] = \nabla f(\mathbf{x}), \text{ and } \mathbb{E}[\|\mathbf{g}(\mathbf{x})]\|^2] \leq G^2$$

for some $G > 0$.

Example (large-scale opt.). Given dataset $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$, ERM optimizes

$$\min_{h\in\mathcal{H}} \sum_{i=1}^{m} \ell(h(\mathbf{x}_i), y_i) \implies$$

full gradient computation requires a pass of ***all data***

stochastic method only uses a ***mini batch*** at each round

Stochastic Gradient Descent

- Consider the following ***convex optimization*** problem:

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$$

Algorithm 2 Stochastic Gradient Descent

Input: noisy gradient oracle $\mathbf{g}(\cdot)$, step sizes $\{\eta_t\}$

1: **for** $t = 1, \ldots, T$ **do**
2: Obtain noisy gradient $\mathbf{g}(\mathbf{x}_t)$
3: Update the model $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \mathbf{g}(\mathbf{x}_t)]$
4: **end for**
5: **return** $\bar{\mathbf{x}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$

$$\mathbb{E}[\mathbf{g}(\mathbf{x})] = \nabla f(\mathbf{x})$$
$$\mathbb{E}[\|\mathbf{g}(\mathbf{x})]\|^2] \leq G^2$$

Stochastic Gradient Descent

Theorem 7 (Convergence of SGD). *Suppose the domain $\mathcal{X} \subseteq \mathbb{R}^d$ has a diameter $D > 0$, and the noisy gradient oracle is unbiased and variance bounded by G^2. SGD with step size $\eta_t = \frac{D}{G\sqrt{t}}$ guarantees*

$$\mathbb{E}[f(\bar{\mathbf{x}}_T)] \leq \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \frac{3GD}{2\sqrt{T}},$$

where $\bar{\mathbf{x}}_T \triangleq \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$ is the output of the SGD algorithm.

Proof of SGD Convergence

Proof. First, we rephrase SGD from lens of ***online convex optimization***.

To see this, we define linear function $f_t(\mathbf{x}) \triangleq \mathbf{g}_t^\top \mathbf{x}$, where $\mathbf{g}_t = \mathbf{g}(\mathbf{x}_t)$.

> **Claim**: deploying OGD over the online functions $\{f_t(\mathbf{x})\}$ is equivalent to SGD proposed in the earlier page.

OGD:
$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)]$$
$$= \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \mathbf{g}(\mathbf{x}_t)]$$

Algorithm 2 Stochastic Gradient Descent

Input: noisy gradient oracle $\mathbf{g}(\cdot)$, step sizes $\{\eta_t\}$

1: **for** $t = 1, \ldots, T$ **do**
2: Obtain noisy gradient $\mathbf{g}(\mathbf{x}_t)$
3: Update the model $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}[\mathbf{x}_t - \eta_t \mathbf{g}(\mathbf{x}_t)]$
4: **end for**
5: **return** $\bar{\mathbf{x}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$

Proof of SGD Convergence

Proof.

$$\mathbb{E}\left[f(\bar{\mathbf{x}}_T)\right] - f(\mathbf{x}^\star) \leq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}_t)\right] - f(\mathbf{x}^\star)$$

($\mathbf{x}^\star = \arg\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$)
(Jensen's inequality)

$$\leq \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \nabla f(\mathbf{x}_t)^\top (\mathbf{x}_t - \mathbf{x}^\star)\right]$$

(convexity)

$$= \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \mathbf{g}_t^\top (\mathbf{x}_t - \mathbf{x}^\star)\right]$$

(this step is true, but not trivial, see Section 3.3 of this paper)

$$= \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} f_t(\mathbf{x}_t) - f_t(\mathbf{x}^\star)\right]$$

(definition of $f_t(\cdot)$)

$$\leq \frac{\text{Regret}_T}{T}$$

(SGD = OGD over $\{f_t(\cdot)\}$)
(regret bound of OGD)

$$\leq \frac{3GD}{2\sqrt{T}} \qquad \square$$

(regret of OGD algorithm)

> **Theorem 3** (Regret bound for OGD). *Under Assumption 1, 2 and 3, online gradient descent (OGD) with step sizes $\eta_t = \frac{D}{G\sqrt{t}}$ for $t \in [T]$ guarantees:*
>
> $$\text{Regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq \frac{3}{2}GD\sqrt{T}.$$

Stochastic Gradient Descent

Theorem 7 (Convergence of SGD). *Suppose the domain $\mathcal{X} \subseteq \mathbb{R}^d$ has a diameter $D > 0$, and the noisy gradient oracle is unbiased and variance bounded by G^2. SGD with step size $\eta_t = \frac{D}{G\sqrt{t}}$ guarantees*

$$\mathbb{E}[f(\bar{\mathbf{x}}_T)] \leq \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) + \frac{3GD}{2\sqrt{T}},$$

where $\bar{\mathbf{x}}_T \triangleq \frac{1}{T}\sum_{t=1}^T \mathbf{x}_t$ is the output of the SGD algorithm.

- We define the linear functions $f_t(\mathbf{x}) \triangleq \mathbf{g}_t^\top \mathbf{x}$ and run Algorithm 2 on f_t, which depends on the decision $\mathbf{x}_t$.
- This actually reveals that OGD can hold even against ***adaptive adversary***.

Stochastic Gradient Descent

- We define the linear functions $f_t(\mathbf{x}) \triangleq \mathbf{g}_t^\top \mathbf{x}$ and run Algorithm 2 on f_t, which depends on the decision $\mathbf{x}_t$.
- This actually reveals that OGD can hold even against ***adaptive adversary***.

> At each round $t = 1, 2, \cdots$
> (1) the player first picks a model $\mathbf{x}_t \in \mathcal{X}$;
> (2) and simultaneously environments pick an online function $f_t : \mathcal{X} \to \mathbb{R}$;
> (3) the player suffers loss $f_t(\mathbf{x}_t)$, observes some information about f_t and updates the model.

oblivious adversary

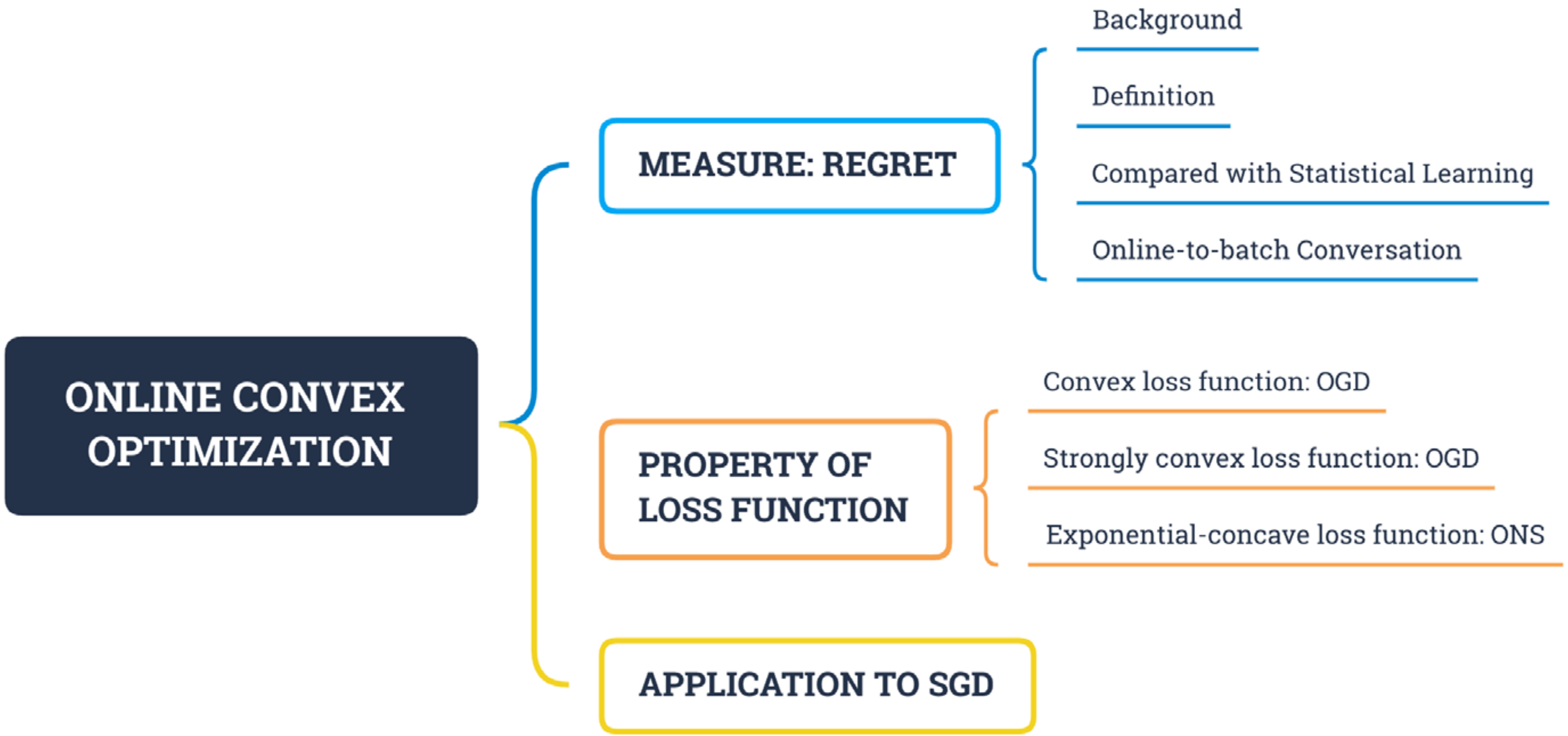
examination

adaptive adversary

interview

- The 'simultaneous' requirement is not necessary in full-info scenario!

Summary



Q & A

Thanks!