Efficient Regret Minimization in Non-Convex Games

Elad Hazan, Karan Singh, Cyril Zhang
Department of Computer Science, Princeton University

Motivation: Non-Convex Games

Repeated games with **non-convex** utility functions. Example: **GAN (generative adversarial network) training**. Two players G and D play a zero-sum game, with continuous strategies θ_G and θ_D, and non-convex objective

$$\ell(\theta_G, \theta_D) := \mathop{\mathbb{E}}_{x \sim \mathcal{D}, z \sim \mathcal{D}'} \left[\log D(x) + \log\left(1 - D(G(z))\right) \right].$$

Also: natural model for irrational players (risk/loss aversion).

Question: What kinds of equilibria are achievable?

Framework: Online Non-Convex Optimization

Each player makes iterative predictions. For round $t = 1, 2, \ldots$

1. Player commits to a decision $\mathbf{x}_t \in \mathcal{K}$, where $\mathcal{K} \subseteq \mathbb{R}^n$ is a convex decision set.
2. Nature presents a smooth **non-convex** loss function $f_t : \mathcal{K} \to \mathbb{R}$.
3. Player suffers loss $f_t(\mathbf{x}_t) \in \mathbb{R}$ for the chosen decision $\mathbf{x}_t$.

Usual goal from online convex optimization [5]: achieve low regret

$$\text{Regret}(T) = \sum_{t=1}^{T} f_t(x_t) - \underbrace{\text{argmin}_{x \in \mathcal{K}} \sum_{t=1}^{T} f_t(x)}_{\text{global min. for fixed decision}} .$$

Global non-convex optimization is computationally **intractable**. We introduce a new framework of **local regret**, with **efficient algorithms** and **tight lower bounds**, generalizing known results in offline and stochastic non-convex optimization.

A Time-Smoothed Regret Measure

Define **local regret** (with window parameter w):

$$\mathfrak{R}_w(T) \overset{\text{def}}{=} \sum_{t=1}^{T} \Bigg\| \underbrace{\nabla \left(\frac{1}{w} \sum_{i=0}^{w-1} f_{t-i} \right)(x_t)}_{\overset{\text{def}}{=} \nabla F_{t,w}(x_t) \text{ time-smoothed gradient}} \Bigg\|^2 .$$

Compare to typical non-convex convergence guarantees [2]:

$$\sum_{t=1}^{T} \|\nabla f(x_t)\|^2 = o(T).$$

Randomized construction gives lower bound $\mathfrak{R}_w(T) \geq \Omega(T/w^2)$.

Time-Smoothed Algorithms

To minimize local regret, take gradient (or second-order) steps on **time-smoothed losses** $F_{t,w}(x) = \frac{1}{w} \sum_{i=0}^{w-1} f_{t-i}(x)$.

Algorithm 1: Time-Smoothed Online Gradient Descent

for $t = 1, \ldots, T$ **do**
  Predict x_t. Observe f_t.
  Update $x_{t+1} \leftarrow x_t - \eta \nabla F_{t,w}(x_{t+1})$.
end for

Algorithm 2: Time-Smoothed Follow-The-Leader

for $t = 1, \ldots, T$ **do**
  Predict x_t. Observe f_t. Initialize $x_{t+1} := x_t$.
  while $\|\nabla F_{t,w}(x_{t+1})\| > \delta$ **do**
    Update $x_{t+1} \leftarrow x_{t+1} - \eta \nabla F_{t,w}(x_{t+1})$.
  end while
end for

Algorithm 3: Time-Smoothed Second-Order Method

for $t = 1, \ldots, T$ **do**
  Predict x_t. Observe f_t. Initialize $x_{t+1} := x_t$.
  while $\|\nabla F_{t,w}(x_{t+1})\| > \delta_1$ or $\nabla^2 F_{t,w}(x_{t+1}) \prec -\delta_2 I$ **do**
    Compute $(\lambda, v) :=$ MinEig $\left(\nabla^2 F_{t,w}(x_{t+1})\right)$.
    Let $g_{t+1} := x_{t+1} - \eta_1 \nabla F_{t,w}(x_{t+1})$. *(gradient step)*
    Let $h_{t+1} := x_{t+1} \pm \eta_2 \lambda v$. *(Hessian steps)*
    Take whichever step makes the most progress on $F_{t,w}$.
  end while
end for

Local Regret Bounds and Offline Reductions

Upper bounds for local regret (with optimal parameters δ, η):

- **Algorithm 1** achieves $\mathfrak{R}_w(T) \leq O(T/w)$, with $O(Tw)$ gradient oracle calls.
- **Algorithm 2** achieves (optimal) $\mathfrak{R}_w(T) \leq O(T/w^2)$. With acceleration by SVRG [1], requires $O(Tw^{5/3})$ gradient oracle calls.
- **Algorithm 3** achieves $\mathfrak{R}_w(T) \leq O(T/w^2)$, while escaping second-order saddle points.

Online framework generalizes offline convergence:

- Guarantees from **Algorithm 2** recovers standard $O(1/\varepsilon)$ convergence rate of GD.
- Guarantees from **Algorithm 1** implies $O(\sigma^4/\varepsilon^2)$ convergence rate of SGD. (Optimal dependence on ε but not σ^2.)
- Reductions are **black-box** (like in online convex optimization).

Solution Concept for Non-Convex Games

Answer: "Smoothed local equilibrium", if players minimize local regret.

$$\forall i \in [k], \left\| \nabla_i \left[\frac{1}{w} \sum_{\tau=0}^{w-1} \text{Utility}_i \left(x_i^{(t)}, x_{-i}^{(t-\tau)} \right) \right] \right\| \leq \varepsilon.$$

Solution concept in repeated non-convex games:

- No player gains too much by deviating locally from their recommended play...
- Provided that everyone else plays strategies sampled uniformly from the past w iterations.

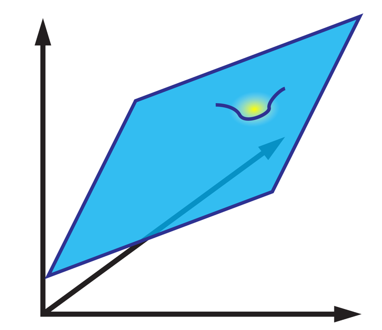
Black-box guarantee: online algorithms minimizing local regret achieve an approximate smoothed local equilibrium, with

$$\varepsilon \leq \sqrt{\frac{k \cdot \mathfrak{R}_w(T)}{T-w}} \overset{\text{Alg. 2}}{\lesssim} O\left(\frac{\sqrt{k}}{w} \right).$$

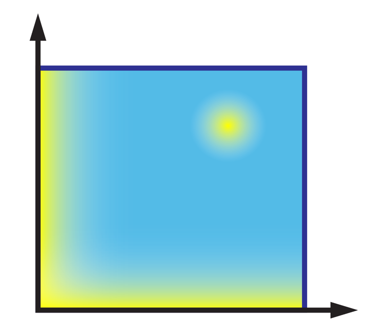
In GAN training, our time-smoothed algorithms are better known as **experience replay**, a known technique for promoting stability. [4]

Note: Projected Gradients for Constraint Sets

Constrained non-convex optimization can be **information-theoretically hard**. To handle constraints, our algorithms use projected gradient descent, and achieve bounds on the **projected gradient**, the vector $\nabla_{\mathcal{K},\eta} f(x)$ such that $x_{t+1} \leftarrow x_t - \eta \nabla_{\mathcal{K},\eta} f(x_t)$ is a *projected gradient descent* step.



Finding a small ∇f is intractable...



...but PGD finds a small $\nabla_{\mathcal{K},\eta} f$.

References

[1] Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. *ICML*, 2016.
[2] S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 2013.
[3] I. Goodfellow et al. Generative adversarial nets. *NIPS*, 2014.
[4] D. Pfau and O. Vinyals. Connecting generative adversarial networks and actor-critic methods. ArXiv preprint, 2016.
[5] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. *ICML*, 2003.