Solving Saddle Point Problems When One of the Dimensions is Small

Egor Gladin
Research Advisor: Alexander Gasnikov

Moscow Institute of Physics and Technology,
Skolkovo Institute of Science and Technology

Optimization Without Borders

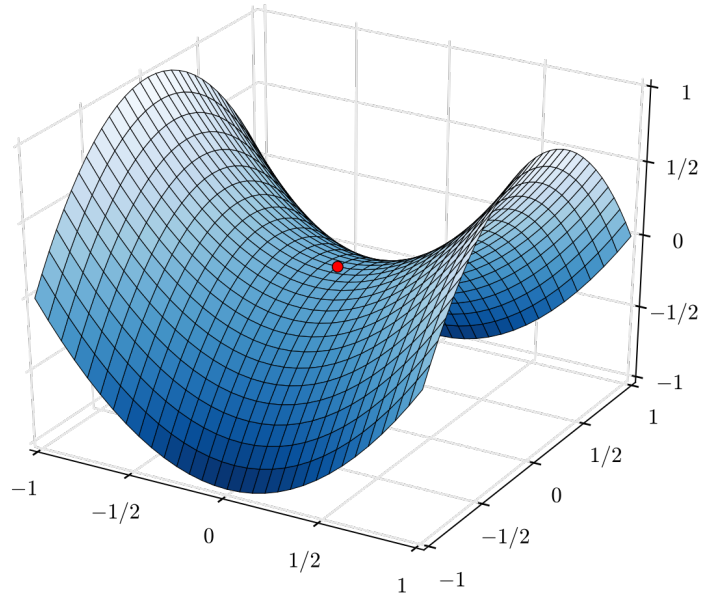
Saddle point (a.k.a. min-max) problems are optimization problems of the form

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y).$$

f is often assumed to be convex-concave.

Solution **(saddle point):**

$$(x_*, y_*) \in \mathcal{X} \times \mathcal{Y} : f(x_*, y) \leqslant f(x_*, y_*) \leqslant f(x, y_*)$$



wikipedia.org/wiki/Saddle_point/

Applications

- machine learning (e.g. GANs);
- computer graphics;
- game theory;
- optimal transport theory.

$$\min_{x \in \mathcal{X}} \max_{y \in \mathbb{R}^{n_y}} f(x, y), \quad \text{where} \tag{1}$$

- $\mathcal{X} \subset \mathbb{R}^{n_x}$ is a compact convex set with nonempty interior, n_x **is small**;
- $f(x, y)$ is convex in x and μ-strongly concave in y;
- for all $x \in \mathcal{X}$ the function $f(x, \cdot)$ is L-smooth, i.e.

$$\left\| \nabla_y f(x, y) - \nabla_y f(x, y') \right\|_2 \leqslant L \left\| y - y' \right\|_2 \quad \forall y, y' \in \mathbb{R}^{n_y}.$$

First-order oracles

$\partial_x f$ and $\nabla_y f$ are available.

Mixed oracle

$\partial_x f$ and $f(x, y)$ are available.

Goal of adversarial attack: craft an example $y' = y_0 + y \in \mathbb{R}^{n_y}$ to mislead an ML model.
Optimization problem:

$$\max_{y \in \mathbb{R}^{n_y}} \mathcal{L}\left(y'\right) - \frac{\gamma}{2} \|y\|_2^2 ,$$

where $\mathcal{L}$ is a loss function.

Robust adversarial attack: mislead m models simultaneously.
Optimization problem:

$$\max_{y \in \mathbb{R}^{n_y}} \min_{1 \le i \le m} \mathcal{L}_i\left(y'\right) - \frac{\gamma}{2} \|y\|_2^2 \iff \max_{y \in \mathbb{R}^{n_y}} \min_{w \in \mathcal{P}} \sum_{i=1}^{m} w_i \mathcal{L}_i\left(y'\right) - \frac{\gamma}{2} \|y\|_2^2 ,$$

where $\mathcal{P}$ is a probability simplex.

Consider the function

$$g(x) := \underbrace{\max_{y \in \mathbb{R}^{n_y}} f(x, y)}_{\text{“inner problem”}}, \tag{2}$$

and rewrite the initial min-max problem as follows:

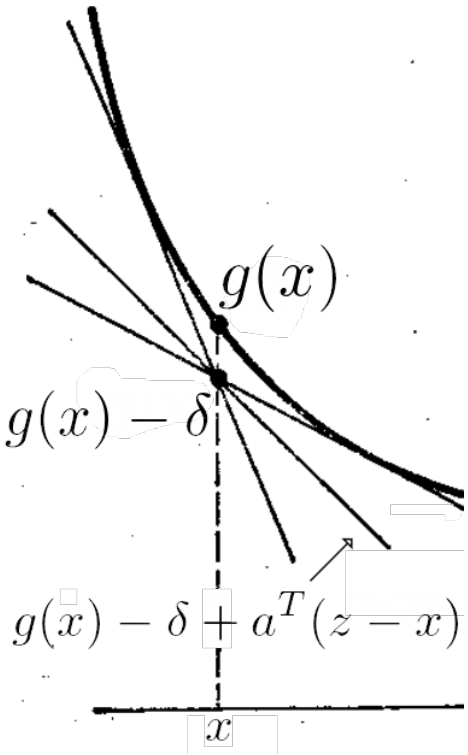
$$\underbrace{\min_{x \in \mathcal{X}} g(x)}_{\text{“outer problem”}}. \tag{3}$$

- Use an iterative method for (3);
- Solve (2) numerically on each iteration to access $g(x)$.

Notion of δ-subgradient

Definition Vector $a \in \mathbb{R}^n$ is called a δ**-subgradient** of a convex function g at x (denoted $a \in \partial_\delta g\,(x)$), if

$$g(z) \geq g(x) + a^\top (z - x) - \delta \quad \forall z \in \operatorname{dom} f.$$



Algorithms: Vaidya's cutting plane method[1]

Consider (possibly **non-smooth**) convex optimizations problem:

$$\min_{x \in \mathcal{X}} g(x). \tag{4}$$

Idea of Vaidya's cutting plane method:

- the algorithm produces a sequence of pairs $\left(A^{(t)}, b^{(t)}\right) \in \mathbb{R}^{m_t \times n} \times \mathbb{R}^{m_t}$ such that the polytope $\left\{x \in \mathbb{R}^n : A^{(t)}x > b^{(t)}\right\}$ contains the solution;
- at each step we either add a constraint to the current polytope to reduce its volume or remove the least "useful" constraint if there are too many of them.

Theorem (Gladin et al.)

After N iterations Vaidya's method with δ-subgradient for the problem (3) returns a point x^N such that

$$g(x^N) - g(x_*) \leqslant O\left(n^{1.5} e^{-\frac{\gamma N}{2n}}\right) + \delta. \tag{5}$$

where $\gamma > 0$ is an algorithm parameter.

[1] Vaidya, "A new algorithm for minimizing convex functions over convex sets"

Smooth convex problem:

$$\min_{y \in \mathbb{R}^n} f(y). \tag{6}$$

For some $\tau > 0$ and $y \in \mathbb{R}^n$, define

$$\operatorname{grad}_f(y, \tau, \mathbf{e}) = \frac{n}{\tau} \left(f(y + \tau \mathbf{e}) - f(y) \right) \mathbf{e}, \tag{7}$$

where $\mathbf{e}$ is uniformly distributed on the unit Euclidean sphere in $\mathbb{R}^n$.

The method produces **three sequences:**

- $y_{k+1} := t_k z_k + (1 - t_k) w_k$
- $w_{k+1} := y_{k+1} - \frac{1}{2L} \mathbf{g}_{k+1}$ with $\mathbf{g}_{k+1} := \operatorname{grad}_f(y_{k+1}, \tau, \mathbf{e}_{k+1})$.
- $z_{k+1} := \arg\min_{z \in \mathbb{R}^n} \left\{ \alpha_{k+1} \langle \mathbf{g}_{k+1}, z - z_k \rangle + \frac{1}{2} \|z_k - z\|_2^2 \right\}$.

[3] Dvurechensky, Gorbunov, and Gasnikov, "An accelerated directional derivative method for smooth stochastic convex optimization"

$$g(x) = \max_{y \in \mathbb{R}^{n_y}} f(x, y); \tag{2}$$

$$\min_{x \in \mathcal{X}} g(x). \tag{3}$$

Proposed approach

Solve the outer problem (3) via Vaidya's method. Solve the inner problem (2) via

- Fast Gradient Method or Varag in the case of 1-order oracles;
- Accelerated Randomized Directional Derivative method for strongly convex functions (ARDDsc) in the case of a mixed oracle.

Theorem

The proposed approach arrives at ε-solution of the problem (3) after

- $N_x = O\left(n_x \log \frac{n_x}{\varepsilon}\right)$ evaluations of $\partial_x f$,
- $N_y = O\left(n_x \sqrt{\frac{L}{\mu}} \log \frac{n_x}{\varepsilon} \log \frac{1}{\varepsilon}\right)$ evaluations of $\nabla_y f$ **or** $n_y \cdot N_y$ evaluations of $f(x, y)$.

Experiment: Robust Adversarial Attacks

$$\min_{w \in \mathcal{P}} \max_{y \in \mathbb{R}^{n_y}} \sum_{i=1}^{m} w_i \mathcal{L}_i(y) - \frac{\gamma}{2} \|y\|_2^2,$$

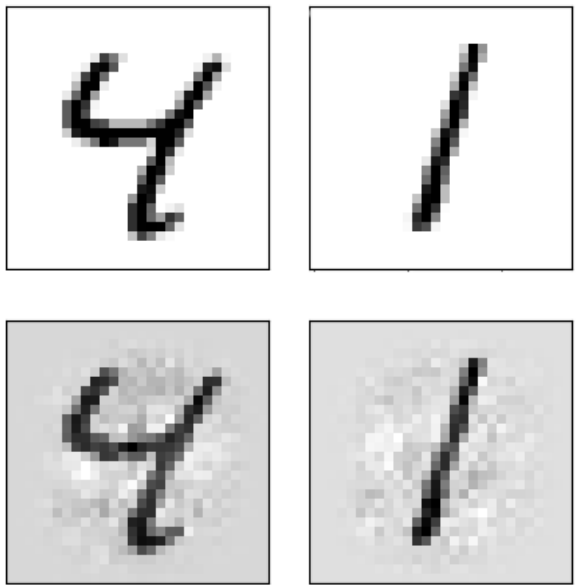
Setup:

- loss function $\mathcal{L}$ is a cross-entropy;
- dataset is MNIST (images of handwritten digits);
- attacked models: 2 models (LogReg and SVM) trained on each half of the dataset, adding up to $m = 4$ models in total.

Attacks caused a sharp fall of accuracy:

Model	Acc_{orig}	$\text{Acc}_{\text{advers}}$
SVM #1	0.94	0.34
SVM #2	0.94	0.36
LogReg #1	0.96	0.44
LogReg #2	0.96	0.44

Original (top) and adversarial (bottom) examples:



Conclusions

Proposed approach for saddle point problems:

1. Requires **one of the dimensions** to be **small**;
2. **Does not** require smoothness/strong convexity in the respective variable;
3. Considers two cases: first-order oracles and **mixed oracle**;
4. Complexity depends (poly-) logarithmically on ε^{-1}.

Thank you for your attention!