Differentially Private Optimization for Smooth Nonconvex ERM

Changyu Gao and Stephen Wright

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

Summary

- We develop simple differentially private optimization algorithms that move along directions of (expected) descent to find an **approximate second-order solution** for nonconvex ERM.
- We use **line search, mini-batching, and a two-phase strategy** to improve the speed and practicality of the algorithm.
- Numerical experiments demonstrate the effectiveness of our approaches, **outperforming SOTA** algorithm DPTR.

Preliminaries

- Privacy protection has become a central issue in machine learning algorithms. *Differential privacy* provides a rigorous and popular framework for quantifying privacy.
- **(ε, δ)-Differential Privacy):** A randomized algorithm $\mathcal{A}$ is (ε, δ)-differentially private (DP) if for all neighboring datasets D, D' and for all events S in the output space of $\mathcal{A}$, the following holds

$$\Pr\left(\mathcal{A}(D) \in S\right) \le e^{\varepsilon} \Pr\left(\mathcal{A}(D') \in S\right) + \delta.$$

  There are other variants of DP. We also use ρ-zCDP in our paper.
- **(ϵ_g, ϵ_H)-2S**: A solution that satisfies approximate second-order solution

$$\|\nabla f(w)\| \le \epsilon_g, \quad \lambda_{\min}\left(\nabla^2 f(w)\right) \ge -\epsilon_H. \tag{1}$$

- **Gaussian Mechanism:** We can achieve differential privacy by adding a Gaussian noise to the output.
- **Problem (DP-ERM):** Find a (ϵ_g, ϵ_H)-2S of the ERM in $\mathcal{R}^d$,

$$f(w) = \frac{1}{n} \sum_{i=1}^{n} l\left(w, x_i\right). \tag{2}$$

- We assume boundedness up to second order and Lipschitz Hessians.

Algorithm

We develop our based on a simple second-order algorithm [2]:

- When gradients are large, take a gradient step.
- When gradients are small, we check the Hessian. If there is a negative direction, we take a negative curvature step. Otherwise, we are done.

We add noise to gradients and Hessians to achieve differential privacy. However, the step sizes and noise parameters need to be chosen carefully to ensure privacy and convergence at the same time.

Algorithm DP Optimization with Second-Order Guarantees

Given: iteration min decrease MIN_DEC, tolerances ϵ_g, ϵ_H, noise parameters σ_f, σ_g, σ_H

Initialize w_0 and sample $z \sim \mathcal{N}(0, \Delta_f^2 \sigma_f^2)$
Compute an upper bound of the required number of iterations as follows

$$T = \left\lceil \frac{f(w_0) + |z| - \underline{f}}{\text{MIN\_DEC}} \right\rceil \tag{3}$$

Choose σ_g and σ_H using T
for $k = 0, 1, \ldots, T-1$ **do**
  Sample $\varepsilon_k \sim \mathcal{N}\left(0, \Delta_g^2 \sigma_g^2 I_d\right)$ and compute the perturbed gradient $\tilde{g}_k = g_k + \varepsilon_k$
  if $\|\tilde{g}_k\| > \epsilon_g$ **then**
    Choose $\gamma_{k,g}$ and set $w_{k+1} \leftarrow w_k - \gamma_{k,g} \tilde{g}_k$ ▷ **Gradient step**
  else
    Sample E_k such that E_k is a $d \times d$ symmetric matrix in which each entry on and above its diagonal is i.i.d. as $\mathcal{N}\left(0, \Delta_H^2 \sigma_H^2\right)$
    Compute perturbed Hessian $\tilde{H}_k = H_k + E_k$
    Compute the minimum eigenvalue of $\tilde{H}_k$ and the corresponding eigenvector $(\tilde{\lambda}_k, \tilde{p}_k)$ satisfying $\|\tilde{p}_k\| = 1$ and $(\tilde{p}_k)^T \tilde{g}_k \le 0$
    if $\tilde{\lambda}_k < -\epsilon_H$ **then**
      Choose $\gamma_{k,H} > 0$ and set $w_{k+1} \leftarrow w_k + \gamma_{k,H} \tilde{p}_k$ ▷ **Negative curvature step**
    else
      return w_k
    end if
  end if
end for

Theorem (informal). Under proper choices of parameters, with probability at least $\{(1 - \zeta/T)(1 - C \exp(-C_1 C d))\}^T$, the algorithm is ρ-zCDP, and outputs a $((1 + c_1)\epsilon_g, (1 + c)\epsilon_H)$-2S, provided that $n \ge n_{\min}$, where the asymptotic dependence of $n_{\min}$ on (ϵ_g, ϵ_H), ρ and d, is

$$n_{\min} = \frac{\sqrt{d}}{\sqrt{\rho}} \tilde{O}\left(\max\left(\epsilon_g^{-2}, \epsilon_g^{-1}\epsilon_H^{-2}, \epsilon_H^{-7/2}\right)\right). \tag{4}$$

Line search and other enhancements

Line search: Instead of using "short steps", we can use line search to choose step sizes for a speedup. We use the *sparse vector technique* [1] to do this without spending too much privacy budget.

Mini-batching: We can carefully choose the parameters to derive a mini-batch version of the algorithm.

Two-phase strategy: Our choice of parameters is based on the worst-case analysis. We can try more aggressive parameters and fall back to conservative estimates if needed.

Eigenvalue computation without forming the full Hessian: We can use the Lanczos method to find an approximation to the minimum eigenvalue and eigenvector, in place of a direct eigenvalue computation.

Experiments

Setting: Covertype dataset, logistic loss with nonconvex regularizer. We experiment under different levels of privacy budget measured by ε.

Covertype: finding a loose solution, $(\epsilon_g, \epsilon_H) = (0.060, 0.245)$

method	$\varepsilon = 0.2$		$\varepsilon = 0.6$		$\varepsilon = 1.0$	
	final loss	runtime	loss	runtime	loss	runtime
TR	0.729 ± 0.028	10.1 ± 9.9	0.729 ± 0.026	8.3 ± 8.6	0.729 ± 0.026	9.5 ± 9.1
TR-B	0.729 ± 0.029	2.2 ± 2.0	0.728 ± 0.027	2.2 ± 2.4	0.729 ± 0.028	2.5 ± 2.4
OPT	0.581 ± 0.057	$\times$	0.712 ± 0.018	0.6 ± 0.2	0.712 ± 0.017	0.5 ± 0.2
OPT-B	0.712 ± 0.018	3.1 ± 2.9	0.712 ± 0.018	3.2 ± 3.0	0.712 ± 0.018	2.9 ± 2.9
OPT-LS	0.577 ± 0.032	$\times$	0.687 ± 0.028	$\mathbf{0.4 \pm 0.1}$	0.699 ± 0.018	$\mathbf{0.4 \pm 0.1}$
2OPT	0.626 ± 0.078	$\times$	0.712 ± 0.017	0.6 ± 0.2	0.712 ± 0.018	0.6 ± 0.2
2OPT-B	0.712 ± 0.018	1.4 ± 0.3	0.712 ± 0.018	1.4 ± 0.4	0.712 ± 0.018	2.0 ± 1.7
2OPT-LS	0.699 ± 0.018	$\mathbf{0.5 \pm 0.2}$	0.699 ± 0.018	0.5 ± 0.2	0.699 ± 0.018	0.5 ± 0.2

Covertype: finding a tight solution: $(\epsilon_g, \epsilon_H) = (0.030, 0.173)$

method	$\varepsilon = 0.2$		$\varepsilon = 0.6$		$\varepsilon = 1.0$	
	final loss	runtime	loss	runtime	loss	runtime
TR	0.516 ± 0.005	$\times$	0.607 ± 0.007	99.6 ± 32.2	0.607 ± 0.005	90.8 ± 21.6
TR-B	0.517 ± 0.005	$\times$	0.603 ± 0.005	32.6 ± 7.9	0.607 ± 0.003	33.4 ± 14.4
OPT	0.506 ± 0.001	$\times$	0.535 ± 0.015	$\times$	0.592 ± 0.003	1.8 ± 0.5
OPT-B	0.597 ± 0.003	$\mathbf{1.3 \pm 0.3}$	0.597 ± 0.003	1.3 ± 0.2	0.597 ± 0.003	1.4 ± 0.3
OPT-LS	0.525 ± 0.009	$\times$	0.527 ± 0.009	$\times$	0.549 ± 0.006	$\times$
2OPT	0.502 ± 0.001	$\times$	0.513 ± 0.003	$\times$	0.519 ± 0.003	$\times$
2OPT-B	0.597 ± 0.003	2.1 ± 0.4	0.597 ± 0.003	2.3 ± 0.5	0.597 ± 0.003	2.3 ± 0.6
2OPT-LS	0.577 ± 0.008	2.1 ± 1.0	0.591 ± 0.001	$\mathbf{0.6 \pm 0.1}$	0.591 ± 0.001	$\mathbf{0.8 \pm 0.2}$

Note: TR: SOTA DP-TR, "-B": mini-batching variant, "OPT": proposed algorithm, "2OPT": algo + two-phase strategy, "-LS": line-search variant.

Our **proposed algorithm runs much faster** than the SOTA algorithm DP-TR. Line search and mini-batching improve upon the short step algorithm, especially when combined with our two-phase strategy. **2OPT-LS consistently performs well** across different settings of parameters.

[1] Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, August 2014.

[2] Stephen J. Wright and Benjamin Recht. *Optimization for Data Analysis*. Cambridge University Press, Cambridge, 2022.