Formatting Instructions For NeurIPS 2022

Xiaoxuan Yu
College of Chemistry and Molecular Engineering
Peking University
Beijing, China
xiaoxuan_yu@pku.edu.cn

Abstract

The abstract paragraph should be indented ½ inch (3 picas) on both the left- and right-hand margins. Use 10 point type, with a vertical spacing (leading) of 11 points. The word **Abstract** must be centered, bold, and in point size 12. Two line spaces precede the abstract. The abstract must be limited to one paragraph.

1 Numerical Experiments

As the final part of our report, we implement the algorithms in the original article with `Julia` and its package `DifferentialEquations.jl` [1]. By comparing ODE direct discretizating (DD) methods described in the article against gradient descent (GD) and Nesterov's accelerated gradient (NAG) methods, we con verify the main results in the theoretical part. The code of these experiments can be found here: `https://github.com/xiaoxuan-yu/Direct-Runge-Kutta-Discretization-Achieves-Acceleration-PKU`.

Inspired by the numerical results by Wilson, Mackey, and Wibisono [2], we generate normal distributed separable dataset and fit a linear model $Ax = b$. Then, we minimize three different kinds of loss functions:

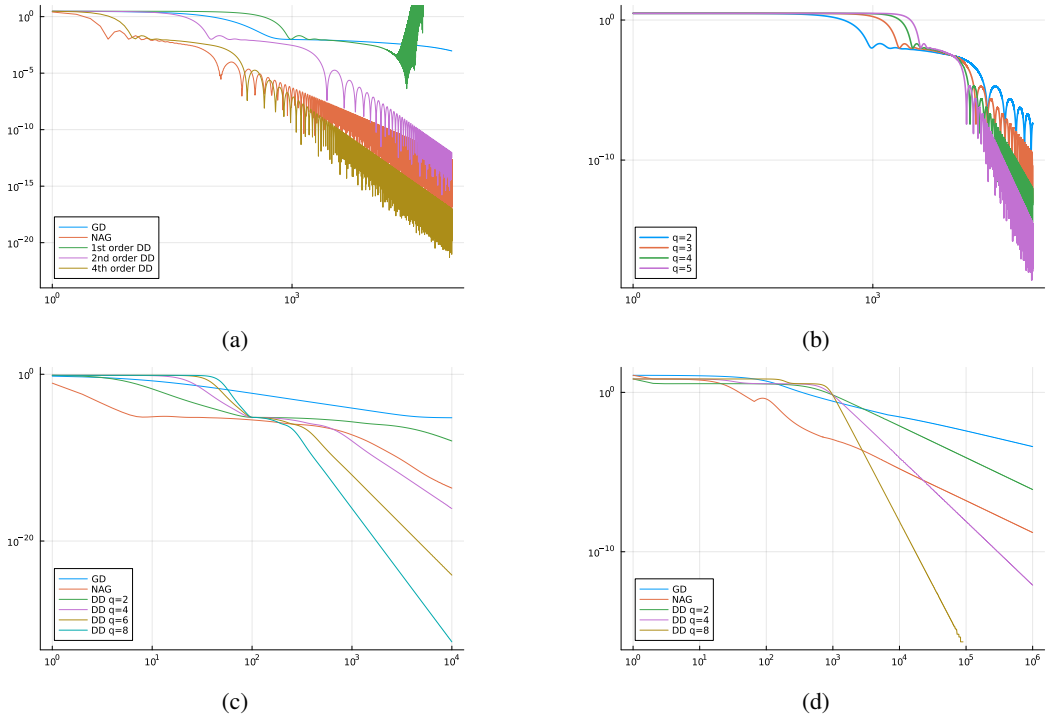
$$\begin{aligned} f_1(x) &= \|Ax - b\|_2^2 \\ f_2(x) &= \sum_i \log(1 + e^{-w_i^{\mathrm{T}} x y_i}) \\ f_3(x) &= \frac{1}{4}\|Ax - b\|_4^4 \end{aligned} \tag{1}$$

where $f_1(\cdot), f_2(\cdot), f_3(\cdot)$ are L_2 loss, logistic loss and L_4 loss, respectively. For each test case and optimization algorithm, we empirically select the learning rate as the largest step length among $\{10^{-k} | k \in \mathbb{Z}\}$ that the method remains stable during the optimization process. Main results are shown in Figure 1 where all figures are on log-log scale.

First, we explore the optimization path of a quadratic function, the L_2 loss, w.r.t. iteration. In particular, we labeled half of the generated data by 0 and the rest by 1. In Figure 1a, the ODE is discretized for $p = 2$ with different Runge-Kutta integrators with $s \in \{1, 2, 4\}$ and compared against GD and NAG algorithm. We can find that except the integrator with $s = 1$ can not converge due to the unstability of the differential format itself, the DD methods shows superiority over GD. By using higher order iterator, the local acceleration is achieved and 4th order DD even converges faster than NAG (although for each iteration, it is obviously more costly than NAG). In Figure 1b, we explore the effect of q is the ODE. Since in the article p keeps the same as the one in the assumption, thus we denotes q the true parameter used in the ODE as below

$$\ddot{x}(t) + \frac{2q+1}{t}\dot{x}(t) + q^2 t^{q-2}\nabla f(x(t)) = 0.$$

Preprint. Under review.



(a) (b) (c) (d)

Figure 1: Experimental results comparing DD with GD and NAG. (a) Convergence path of GD, NAG and DD with different Runge-Kutta integrators of degree $s = 1, 2, 4$ on L_2 loss. (b) The optimization of L_2 loss by DD with different choices of q values with 4-th order Runge-Kutta integrator RK4. (c) Minimization of L_4 loss by GD, NAG and DD with different q vaues with a 2-nd order Runge-Kutta integrator. (d) Minimization of logistic loss by GD, NAG and DD with different q vaues with a 4-th order Runge-Kutta integrator.

We optimize the same L_2 loss with different values of q. By selecting smaller learning rates and increasing the numerical precision by using longer floats, the phenomenon that DD method diverges when $q > 2$ is not observed. Instead, we found that for $q \in \{2, 3, 4, 5\}$, larger q will give out faster convergence.

Then the minimization of L_4 loss (Figure 1c) and logistic loss (Figure 1d) is studied. We use 2-nd order Runge-Kutta integrator SSPRK22 for logistic loss optimization and 4-th order Runge-Kutta integrator RK4 for L_4 loss. As shown in Figure 1c and 1d, the loss decrease faster for larger q, as we can observed in above experiment about L_2 loss.

References

[1] Christopher Rackauckas and Qing Nie. "Differentialequations.jl–a performant and feature-rich ecosystem for solving differential equations in julia". In: *Journal of Open Research Software* 5.1 (2017).

[2] Ashia C Wilson, Lester Mackey, and Andre Wibisono. "Accelerating Rescaled Gradient Descent: Fast Optimization of Smooth Functions". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: `https://proceedings.neurips.cc/paper/2019/file/7a2b33c672ce223b2aa5789171ddde2f-Paper.pdf`.