Qeios

Peer Review

Review of: "A Quasilinear Algorithm for Computing Higher-Order Derivatives of Deep Feed-Forward Neural Networks"

Colby Fronk[1]

1. University of California, Santa Barbara, United States

I recommend this paper for publication after revisions.

1. Below is a paper that explores a semi-recent method for improving the computational efficiency of higher-order derivatives:

   Bettencourt, J., Johnson, M. J. & Duvenaud, D. Taylor-mode automatic differentiation for higher-order derivatives in JAX. in *Conf. Neural Inform. Process. Syst.* (2019).

   I suggest including a mention of this method in your paper and outlining situations where each method might excel relative to the other. Additional computational comparisons for the forward and backward evaluation components of the paper would be an excellent enhancement for the scientific community and add significant merit to your work, though I recognize that this may not be feasible for the PINN portion of the results.
2. I recommend only including the log scale part of the graph for figures 1, 2, 3, etc.
3. In your current setup, Figures 1-5 are using stars for every data point. You might consider using distinct shapes for each curve to improve readability, especially when printed in greyscale.
4. In your paper, you mention that “the performance gains afforded by -TANGENTPROP decrease as we increase the batch size” and note some issues with the batch dimension and vectorization. You might want to explore using JAX in the future, as it often deals with vectorization better than PyTorch. That said, it’s something to keep in mind for later work rather than a required update for this paper.

 doi.org/10.32388/XUEZRC

Declarations

Potential competing interests: No potential competing interests to declare.