Differentiability of Sim Pieces

Sean Gasiorowski

On behalf of the neutrino ML team

March 23rd, 2022

U.S. DEPARTMENT OF ENERGY | Stanford University

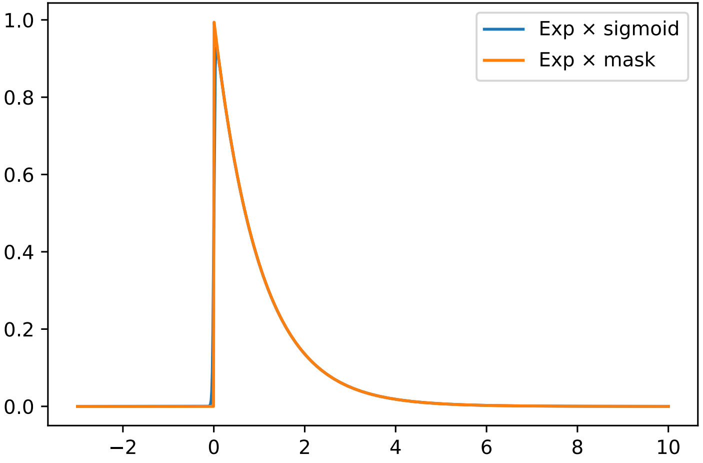
SLAC NATIONAL ACCELERATOR LABORATORY

Long Diff and Differentiability

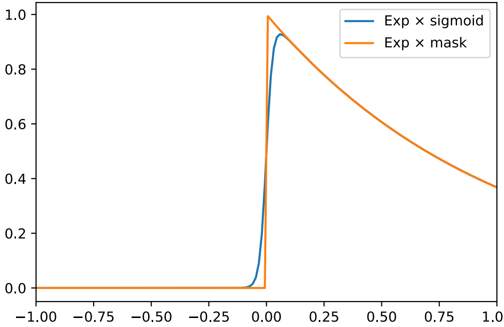
- Have been going piece by piece through the code to find where gradients break

- For long_diff — notable spot is truncated exponential (used for current model)
  - Sharp mask for x < 0 => poorly behaved gradients
    - x-axis ~ time t when we evaluate current
    - t_0 = time of arrival at anode
    - Induced current ~exponential fall off after arrival ($t > t_0$)
    - $t < t_0$ no current (hasn't arrived yet)

  - Try: softening a bit with a sigmoid



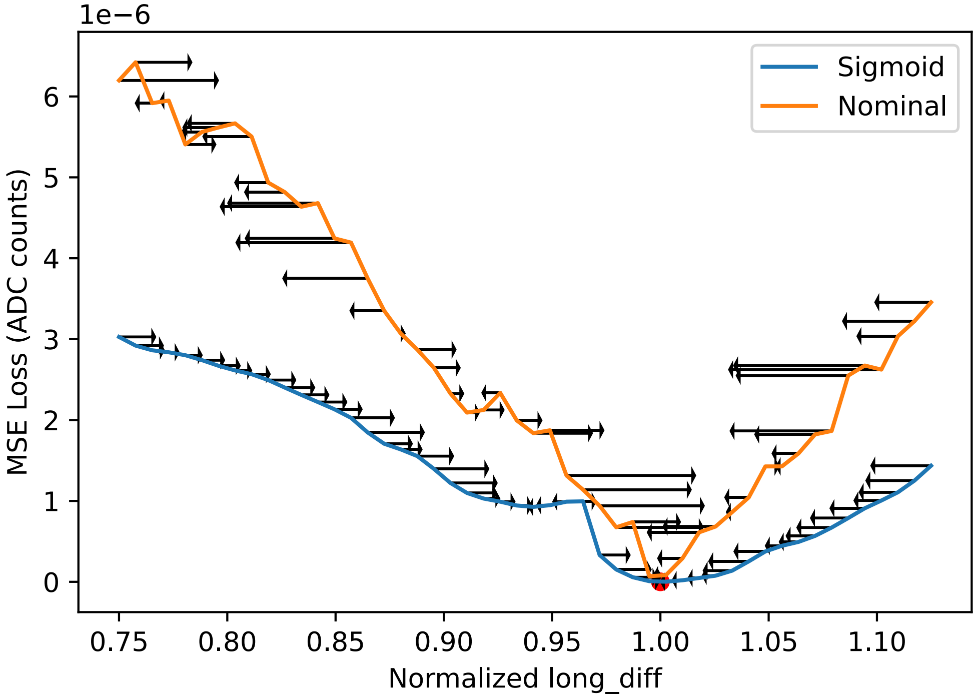
Example (and zoom) for sigmoid with growth rate 70 (sigmoid(70*x)) vs the hard cutoff mask



Long Diff and Differentiability

- Loss curves masking with sigmoid(x*70), with -1e3*gradient shown in arrows for each point

- Sigmoid (blue) smoother, gradients pointed in a reasonable direction

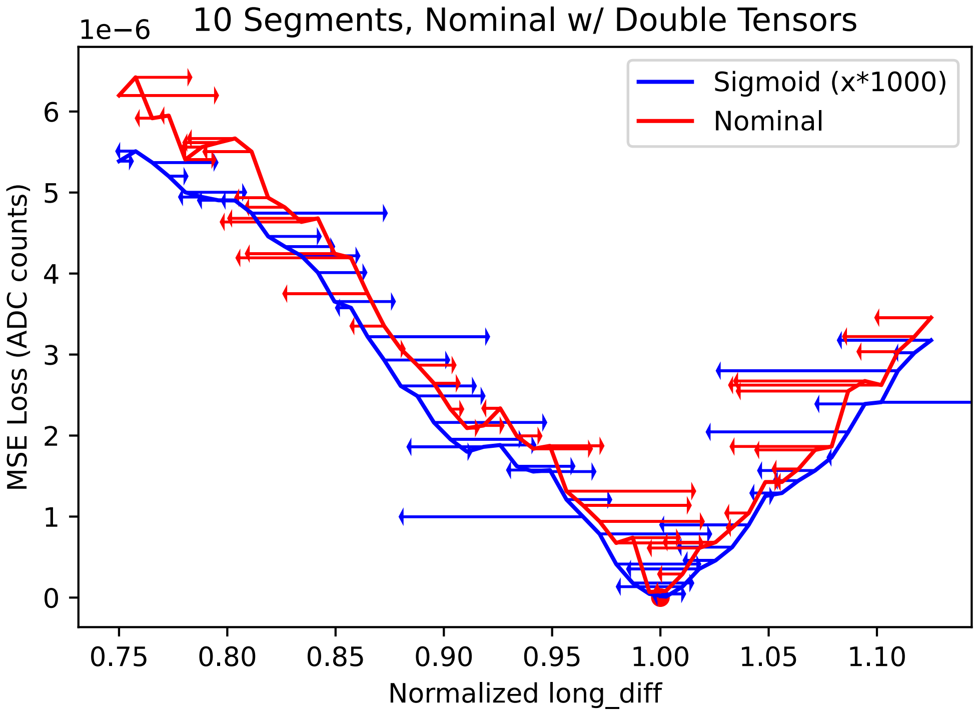


Long Diff and Differentiability

- Increasing sigmoid sharpness (x*1000) in blue
  - Approaches sharp mask

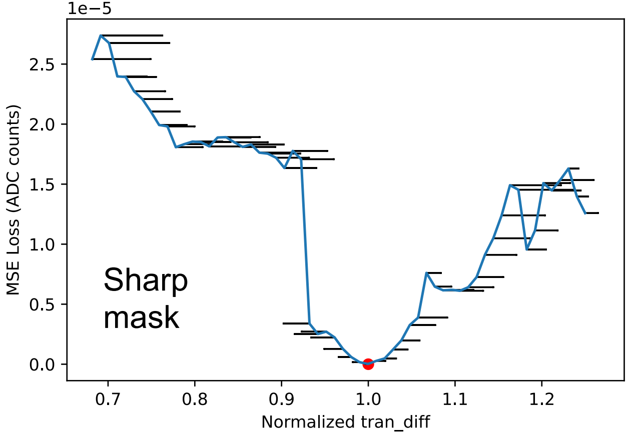
  - Gradients still seem reasonable, but curve is bumpier

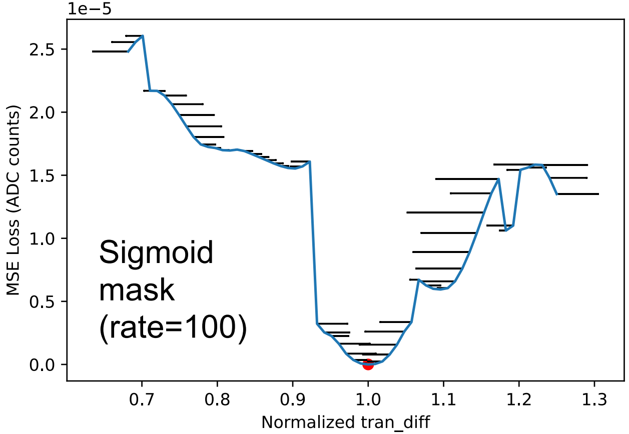
- Ideas (discussions with Daniel, Youssef):
  - Optimize with smoothed sim, tune sigmoid width to approach sharp cutoff as training progresses
    - To check: if compare smoothed to "sharp" sim, what does loss like like



Other Parameters

- Noticed this in long_diff, but impact seen for other parameter gradients
  - tran_diff shown on right — only two track segments as a test
  - Gradients much more sensible with sigmoid mask





Conclusions

- Things like arange break gradient flow
  - But even with an intact computational graph, gradients can be nasty!

- To check:
  - 1d loss surfaces for eField, lifetime, Ab, kb, MeVToElectrons, and now long_diff look ~reasonable in a noiseless case (plots forthcoming, have seen some examples from Yifan)
    - Need to look more at: vdrift, tran_diff

  - Impact of these sharpness effects vs coverage/statistics

  - How do we tune smoothness vs accuracy?

  - Once 1d noiseless look ok
    - Impact of readout noise
    - Come back to multi-parameter fits