Can Layer Normalization Destroy All Gradients?

Due: Wednesday, October 21 at 11:59pm

Instructions: Please submit your *handwritten* solutions as a single PDF on Gradescope.

Layer normalization transforms $\mathbf{z} \in \mathbb{R}^d$ by

$$N(\mathbf{z}) = \frac{\mathbf{z} - \mu\mathbf{1}}{\sigma}, \qquad \mu = \frac{1}{d}\sum_{i=1}^{d} z_i, \qquad \sigma^2 = \frac{1}{d}\sum_{i=1}^{d}(z_i - \mu)^2.$$

Here $\mathbf{1}$ is the vector of ones. Assume $\sigma > 0$ and omit the small constant usually added to the variance for numerical stability.

1. Suppose $d = 2$, so $\mathbf{z} = (a, b)$ with $a \neq b$. Find $N(\mathbf{z})$.
2. The Jacobian describes how the normalized output changes with the input. Show that it is the zero matrix whenever $d = 2$ and $a \neq b$.
3. Now allow the input to have more coordinates. For general $d \geq 2$, show that the Jacobian of N has rank at most $d - 2$.

Hint: For part 3, what happens when you add the same constant to every coordinate, or multiply every coordinate by a positive constant?

Takeaway: *Normalization removes changes in the common offset and scale. With just two coordinates, it keeps only their ordering, so small input changes have no effect and no gradient passes through. In higher dimensions, these same two directions are still lost.*