CS 316: Introduction to Deep Learning

Numerical Stability and Initialisation

Week 6

Dr Abdul Samad

Lecture Outline

- Vanishing and exploding gradients
- Breaking Symmetry
- Xavier initialization

Gradient Vanishing and Exploding

- Consider a deep network with $\mathbf{L}$ layers, input $\mathbf{x}$ and output $\mathbf{o}$.
- With each layer l being defined by a transformation f_l parameterized by weights $\mathbf{W}^{(l)}$ and whose hidden layer output is $\mathbf{h}^{(l)}$ (let $\mathbf{h}^{(0)} = \mathbf{x}$), our network can be expressed as:

$$\mathbf{h}^{(l)} = f_l(\mathbf{h}^{(l-1)}) \text{ and thus } \mathbf{o} = f_L \circ \ldots \circ f_1(\mathbf{x}).$$

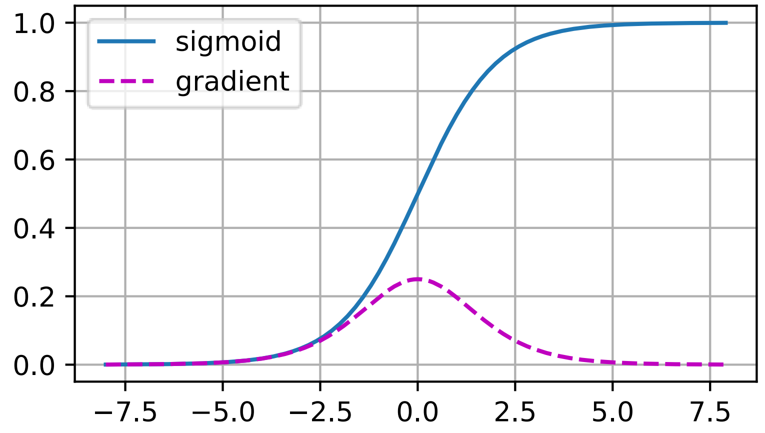
- If all the hidden layer outputs and the input are vectors, we can write the gradient of $\mathbf{o}$ with respect to any set of parameters $\mathbf{W}^{(l)}$as follows:

$$\partial_{\mathbf{W}^{(l)}} \mathbf{o} = \underbrace{\partial_{\mathbf{h}^{(L-1)}} \mathbf{h}^{(L)}}_{\mathbf{M}^{(L)} \stackrel{\text{def}}{=}} \cdot \ldots \cdot \underbrace{\partial_{\mathbf{h}^{(l)}} \mathbf{h}^{(l+1)}}_{\mathbf{M}^{(l+1)} \stackrel{\text{def}}{=}} \underbrace{\partial_{\mathbf{W}^{(l)}} \mathbf{h}^{(l)}}_{\mathbf{v}^{(l)} \stackrel{\text{def}}{=}} .$$

- In other words, this gradient is the product of $L-1$ matrices $\mathbf{M}^{(L)} \cdot \ldots \cdot \mathbf{M}^{(l+1)}$ and the gradient vector $\mathbf{v}^{(l)}$.
- Consequently, we are susceptible to the problem of numerical underflow. In addition to, gradients of unpredictable magnitude threaten the stability of optimization algorithms. We may be facing parameter updates that are excessively large or excessively small, rendering the learning impossible as the parameters hardly move after each update

Gradient Vanishing

- The choice of the activation function that is applied following each layer's linear operations is a common cause of the vanishing gradient issue.
- For example, let’s consider the sigmoid activation function.



- In the graph above, we can see that the sigmoid’s gradient vanishes both when its inputs are large and when they are small.
- Moreover, when backpropagating through many layers, unless the inputs to many of the sigmoid’s are close to zero, the overall product may vanish.
- When our network boasts many layers, unless are careful, the gradient will likely be cut off at some layer.

Figure Source: Sigmoid

Gradient Exploding

- The technique with which weights are initialized is a common cause of the expanding gradient problem.
- For example, let's take an initial matrix M and then 100 Gaussian Random matrices. For the scale that we picked (the choice of the variance $\sigma^2 = 1$), the matrix product explodes.

$$\mathbf{M} = \begin{bmatrix} -0.1882 & 1.4522 & 0.1471 & -1.4465 \\ 0.9311 & -0.4740 & -0.2528 & -0.7200 \\ -1.1675 & -1.2031 & 0.0862 & -0.9560 \\ 0.6061 & 0.7875 & -0.1915 & -0.8064 \end{bmatrix}$$

- After multiplying with 100 Gaussian random matrices the matrix product is equal to:

$$\mathbf{Result} = \begin{bmatrix} 6.9766e+23 & 3.6144e+24 & -1.8983e+24 & 2.5633e+24 \\ -2.0041e+22 & -1.0383e+23 & 5.4530e+22 & -7.3634e+22 \\ -5.2698e+22 & -2.7301e+23 & 1.4339e+23 & -1.9362e+23 \\ 3.8219e+23 & 1.9800e+24 & -1.0399e+24 & 1.4042e+24 \end{bmatrix},$$

Breaking Symmetry

- Another problem in neural network design is the symmetry inherent in their parameterization.

- Assume that we have a single MLP with one hidden and two units.

- In this case, we could permute the weights $\mathbf{W}^{(1)}$ of the first layer and likewise permute the weights of the output layer to obtain the same function. In other words, we have permutation symmetry among the hidden units of each layer.

- Now imagine if we initialized all the parameters of the hidden layer $\mathbf{W}^{(1)} = c$ for some $\mathbf{W}^{(1)}$ constant c. During forward propagation, each hidden unit takes the same inputs and parameters, producing the same activation, which is fed to the output unit. During backpropagation, differentiating the same output unit with respect to the parameters gives a gradient whose elements all take the same value. Thus, after gradient-based iteration, all the elements of $\mathbf{W}^{(1)}$ still take the same value.

- Such iterations would never the break the symmetry on its own. The hidden layer would behave as if it only had a single unit.

Xavier Initialization

- With n_{in} inputs x_j and their associated weights w_{ij} for this layer, an output is given by:

$$o_i = \sum_{j=1}^{n_{\mathrm{in}}} w_{ij} x_j.$$

- The weights w_{ij} are all drawn independently from the same distribution. Furthermore, this distribution has zero mean and variance σ^2.
- The inputs x_j are all drawn independently from the same distribution. Furthermore, this distribution has zero mean and variance γ^2.

$$E[o_i] = \sum_{j=1}^{n_{\mathrm{in}}} E[w_{ij} x_j] = \sum_{j=1}^{n_{\mathrm{in}}} E[w_{ij}] E[x_j] = 0$$

$$\mathrm{Var}[o_i] = E[o_i^2] - (E[o_i])^2 = \sum_{j=1}^{n_{\mathrm{in}}} E[w_{ij}^2 x_j^2] - 0 = \sum_{j=1}^{n_{\mathrm{in}}} E[w_{ij}^2] E[x_j^2]$$

$$\mathrm{Var}[o_i] = n_{\mathrm{in}} \sigma^2 \gamma^2.$$

Xavier Initialization

- One way to keep the variance fixed is to set $n_{in}\sigma^2 = 1$.
- Using the same reasoning as for forward propagations, we see that the gradient's variance can blow up unless $n_{out}\sigma^2 = 1$ where n_{out} is the number of outputs of this layers.
- We cannot possibly satisfy both conditions simultaneously. Consequently, we try to satisfy:

$$\frac{1}{2}(n_{\text{in}} + n_{\text{out}})\sigma^2 = 1 \text{ or equivalently } \sigma = \sqrt{\frac{2}{n_{\text{in}} + n_{\text{out}}}}.$$

- Typically, the Xavier initialization samples weights from a Gaussian distribution with zero mean and variance $\sigma^2 = \frac{2}{n_{\text{in}} + n_{\text{out}}}$.
- We can also adapt Xavier's intuition to choose the variance when sampling weights from a uniform distribution.
- The uniform distribution $U(-a, a)$ has the variance $\frac{a^2}{3}$. Plugging $\frac{a^2}{3}$ in σ^2 gives

$$U\left(-\sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}, \sqrt{\frac{6}{n_{\text{in}} + n_{\text{out}}}}\right).$$