CS 189/289A Introduction to Machine Learning
Fall 2024 Jennifer Listgarten, Saeed Saremi DIS4

1 Backrop in Practice: Staged Computation

For the function $f(x, y, z) = (x + y)z$:

(a) Decompose f into two simpler functions.

(b) Draw the network that represents the computation of f.

(c) Write the forward pass and backward pass (backpropagation) in the network.

(d) Update your network drawing with the intermediate values in the forward and backward pass. Use the inputs $x = -2$, $y = 5$, and $z = -4$.

2 Backpropagation Practice

Disclaimer: the notation used in the following problem is different from the notation used in homework 3. In this discussion, like in discussion 0, we refer to $\frac{\partial \ell}{\partial W}$ as the **derivative** of ℓ with respect to W but we use this same notation in homework 3 to refer to the **gradient** instead. Remember that gradients and derivatives are transposes of each other.

(a) Assume that you have functions $f(x_1, x_2, \ldots, x_n)$, and $g_i(w) = x_i$ for $i = 1, \ldots, n$. Sketch out the computation graph for $f(g_1(w), g_2(w), \ldots, g_n(w))$. How would you compute the following derivative?

$$\frac{\mathrm{d}}{\mathrm{d}w} f(g_1(w), g_2(w), \ldots, g_n(w))$$

(b) Let $w_1, w_2, \ldots, w_n \in \mathbb{R}^d$, and we refer to these weights together as $W \in \mathbb{R}^{n \times d}$. We also have $x \in \mathbb{R}^d$ and $y \in \mathbb{R}$. Consider the function

$$f(W, x, y) = \left(y - \sum_{i=1}^{n} \phi(w_i^\top x + b_i) \right)^2 .$$

Sketch the computation graph for this function.

(c) Suppose $\phi(x)$ (from the previous part) is the sigmoid function, $\sigma(x)$. Compute the derivatives $\frac{\partial f}{\partial w_i}$ and $\frac{\partial f}{\partial b_i}$. Use the computational graph you drew in the previous part to guide you.

(d) Write down a single gradient descent update for $w_i^{(t+1)}$ and $b_i^{(t+1)}$, assuming step size ϵ. You answer should be in terms of $w_i^{(t)}$, $b_i^{(t)}$, x, and y.

(e) Define the cost function

$$\ell(x) = \frac{1}{2}\left\|W^{(2)}\Phi\left(W^{(1)}x + b\right) - y\right\|_2^2, \tag{1}$$

where $W^{(1)} \in \mathbb{R}^{d\times d}$, $W^{(2)} \in \mathbb{R}^{d\times d}$, and $\Phi : \mathbb{R}^d \to \mathbb{R}^d$ is some nonlinear transformation. Compute the derivatives $\frac{\partial \ell}{\partial x}, \frac{\partial \ell}{\partial W^{(1)}}, \frac{\partial \ell}{\partial W^{(2)}}$, and $\frac{\partial \ell}{\partial b}$.

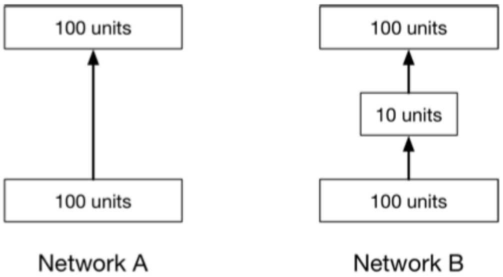
(f) Suppose Φ is the identity map. Write down a single gradient descent update for $W_{t+1}^{(1)}$ and $W_{t+1}^{(2)}$ assuming step size ϵ. Your answer should be in terms of $W_t^{(1)}$, $W_t^{(2)}$, b_t, x, and y.

3 Model Intuition

(a) What can go wrong if you just initialize all the weights in a neural network to exactly zero? What about to the same nonzero value?

(b) Adding nodes in the hidden layer gives the neural network more approximation ability, because you are adding more parameters. How many weight parameters are there in a neural network with architecture specified by $d = \left[d^{(0)}, d^{(1)}, ..., d^{(N)}\right]$, a vector giving the number of nodes in each of the N layers? Evaluate your formula for a 2 hidden layer network with 10 nodes in each hidden layer, an input of size 8, and an output of size 3.

(c) Consider the two networks in the image below, where the added layer in going from Network A to Network B has 10 units with linear activation. Give one advantage of Network A over Network B, and one advantage of Network B over Network A.


100 units
100 units
Network A
100 units
10 units
100 units
Network B