Backpropagation Supplement

CS114B Lab 5

Kenneth Lai

February 17, 2023

Gradients in Feedforward Neural Networks

- We want to compute $\dfrac{\partial L}{\partial W_{jk}^{[i]}}$
- Chain Rule of calculus: $\dfrac{dy}{dx} = \dfrac{dy}{dz}\dfrac{dz}{dx}$
- Looking at the graph: $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \dfrac{\partial L}{\partial a_k^{[i]}}\dfrac{\partial a_k^{[i]}}{\partial z_k^{[i]}}\dfrac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \dfrac{\partial L}{\partial a_k^{[i]}} \dfrac{\partial a_k^{[i]}}{\partial z_k^{[i]}} \dfrac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\frac{\partial L}{\partial W_{jk}^{[i]}} = \frac{\partial L}{\partial a_k^{[i]}} \frac{\partial a_k^{[i]}}{\partial z_k^{[i]}} a_j^{[i-1]}$
    - $\frac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \dfrac{\partial L}{\partial a_k^{[i]}} g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\dfrac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\dfrac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \dfrac{\partial L}{\partial a_k^{[i]}} g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\dfrac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\dfrac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$
      - Let $g'^{[i]}(z_k^{[i]})$ be the derivative of the activation function
      - For the logistic function: $g'^{[i]}(z_k^{[i]}) = a_k^{[i]}(1 - a_k^{[i]})$
      - ...

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\frac{\partial L}{\partial W_{jk}^{[i]}} = \frac{\partial L}{\partial a_k^{[i]}} g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\frac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\frac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$
    - $\frac{\partial L}{\partial a_k^{[i]}} = ?$

Gradients in Feedforward Neural Networks

- Note that for a hidden neuron, $a_k^{[i]}$ is an input to each non-bias neuron ℓ in layer $i+1$

Gradients in Feedforward Neural Networks

- Note that for a hidden neuron, $a_k^{[i]}$ is an input to each non-bias neuron ℓ in layer $i+1$
- Chain Rule of multivariable calculus:

$$\frac{df(g_1(x), \ldots, g_n(x))}{dx} = \sum_{i=1}^{n} \frac{\partial f}{\partial g_i(x)} \frac{dg_i(x)}{dx}$$

Gradients in Feedforward Neural Networks

- Note that for a hidden neuron, $a_k^{[i]}$ is an input to each non-bias neuron ℓ in layer $i+1$
- Chain Rule of multivariable calculus:

$$\frac{df(g_1(x), ..., g_n(x))}{dx} = \sum_{i=1}^{n} \frac{\partial f}{\partial g_i(x)} \frac{dg_i(x)}{dx}$$

- Express L as a function of $z_\ell^{[i+1]}$: $\frac{\partial L}{\partial a_k^{[i]}} = \sum_\ell \frac{\partial L}{\partial z_\ell^{[i+1]}} \frac{\partial z_\ell^{[i+1]}}{\partial a_k^{[i]}}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_{\ell} \dfrac{\partial L}{\partial z_{\ell}^{[i+1]}} \dfrac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\dfrac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\dfrac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$
    - $\dfrac{\partial L}{\partial a_k^{[i]}} = \sum_{\ell} \dfrac{\partial L}{\partial z_{\ell}^{[i+1]}} \dfrac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $$\frac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_{\ell} \frac{\partial L}{\partial z_{\ell}^{[i+1]}} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$$
    - $$\frac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$$
    - $$\frac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$$
    - $$\frac{\partial L}{\partial a_k^{[i]}} = \sum_{\ell} \frac{\partial L}{\partial z_{\ell}^{[i+1]}} \frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}}$$
    - $$\frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}} = W_{k\ell}^{[i+1]}$$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\frac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_{\ell} \delta_{\ell}^{[i+1]} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\frac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\frac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$
    - $\frac{\partial L}{\partial a_k^{[i]}} = \sum_{\ell} \frac{\partial L}{\partial z_{\ell}^{[i+1]}} \frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}}$
    - $\frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}} = W_{k\ell}^{[i+1]}$
    - $\frac{\partial L}{\partial z_{\ell}^{[i+1]}} = \delta_{\ell}^{[i+1]}$

Gradients in Feedforward Neural Networks

- For a hidden neuron:
  - $\frac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_{\ell} \delta_{\ell}^{[i+1]} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$
    - $\frac{\partial z_k^{[i]}}{\partial W_{jk}^{[i]}} = a_j^{[i-1]}$
    - $\frac{\partial a_k^{[i]}}{\partial z_k^{[i]}} = g'^{[i]}(z_k^{[i]})$
    - $\frac{\partial L}{\partial a_k^{[i]}} = \sum_{\ell} \frac{\partial L}{\partial z_{\ell}^{[i+1]}} \frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}}$
    - $\frac{\partial z_{\ell}^{[i+1]}}{\partial a_k^{[i]}} = W_{k\ell}^{[i+1]}$
    - $\frac{\partial L}{\partial z_{\ell}^{[i+1]}} = \delta_{\ell}^{[i+1]}$
      - Let $\delta_{\ell}^{[i+1]}$ be the "error" in neuron ℓ in layer $i+1$
      - What is $\delta_{\ell}^{[i+1]}$?

Backpropagation

- We can compute $\dfrac{\partial L}{\partial W_{k\ell}^{[\mathcal{L}]}} = \dfrac{\partial L}{\partial a_\ell^{[\mathcal{L}]}} \dfrac{\partial a_\ell^{[\mathcal{L}]}}{\partial z_\ell^{[\mathcal{L}]}} \dfrac{\partial z_\ell^{[\mathcal{L}]}}{\partial W_{k\ell}^{[\mathcal{L}]}}$ for an output neuron ℓ in layer $\mathcal{L}$

Backpropagation

- We can compute $\frac{\partial L}{\partial W_{k\ell}^{[\mathcal{L}]}} = \frac{\partial L}{\partial a_\ell^{[\mathcal{L}]}} \frac{\partial a_\ell^{[\mathcal{L}]}}{\partial z_\ell^{[\mathcal{L}]}} \frac{\partial z_\ell^{[\mathcal{L}]}}{\partial W_{k\ell}^{[\mathcal{L}]}}$ for an output neuron ℓ in layer $\mathcal{L}$
- If we have already computed $\frac{\partial L}{\partial W_{k\ell}^{[i+1]}}$ for some neuron ℓ in layer $i+1$, then we have also computed $\delta_\ell^{[i+1]} = \frac{\partial L}{\partial z_\ell^{[i+1]}} = \frac{\partial L}{\partial a_\ell^{[i+1]}} \frac{\partial a_\ell^{[i+1]}}{\partial z_\ell^{[i+1]}}$

Backpropagation

- We can compute $\dfrac{\partial L}{\partial W_{k\ell}^{[\mathcal{L}]}} = \dfrac{\partial L}{\partial a_\ell^{[\mathcal{L}]}} \dfrac{\partial a_\ell^{[\mathcal{L}]}}{\partial z_\ell^{[\mathcal{L}]}} \dfrac{\partial z_\ell^{[\mathcal{L}]}}{\partial W_{k\ell}^{[\mathcal{L}]}}$ for an output neuron ℓ in layer $\mathcal{L}$
- If we have already computed $\dfrac{\partial L}{\partial W_{k\ell}^{[i+1]}}$ for some neuron ℓ in layer $i+1$, then we have also computed $\delta_\ell^{[i+1]} = \dfrac{\partial L}{\partial z_\ell^{[i+1]}} = \dfrac{\partial L}{\partial a_\ell^{[i+1]}} \dfrac{\partial a_\ell^{[i+1]}}{\partial z_\ell^{[i+1]}}$
- We can then use $\delta_\ell^{[i+1]}$ to calculate $\dfrac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_\ell \delta_\ell^{[i+1]} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$ for the previous neurons k in layer i

Backpropagation

- $$\frac{\partial L}{\partial W_{jk}^{[i]}} = \left(\sum_{\ell} \delta_{\ell}^{[i+1]} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]}) a_j^{[i-1]}$$
- $$\frac{\partial L}{\partial b_k^{[i]}} = \left(\sum_{\ell} \delta_{\ell}^{[i+1]} W_{k\ell}^{[i+1]} \right) g'^{[i]}(z_k^{[i]})$$