Notation:

May 19, 2021

Training Set: $\{(x^{(1)}, y^{(1)}) \dots (x^{(m)}, y^{(m)})\}$

L: # of layers in network

s_l: # of units (not counting bias unit) in layer l

For Binary Classification:

$s_L = 1 \rightarrow$ one output unit

$y = 0$ or 1

$h_\theta(x) \in \mathbb{R}$

For Multi-Class Classification:

K classes

$s_L = K \quad (K \geq 3)$

$y \in \mathbb{R}^K$

$h_\theta(x) \in \mathbb{R}^K$

Cost Function:

Logistic Regression:

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

Neural Network:

sums all output nodes

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log(h_\theta(x^{(i)}))_k + (1-y_k^{(i)}) \log(1-(h_\theta(x^{(i)}))_k)\right] + \frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(\theta_{ji}^{(l)})^2$$

sums columns, sums rows

sums all theta matrices θ

This is cross entropy w/ L2 regularization

Gradient Computation:

We want to find $\min J(\theta)$.

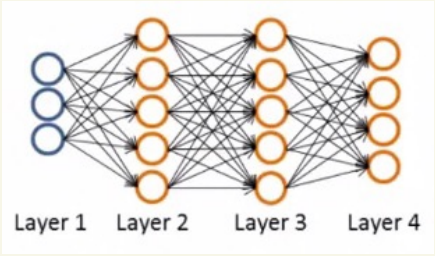
Need to compute:

→ $J(\theta)$

→ $\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta)$, $\theta_{ij}^{(l)} \in \mathbb{R}$

How?

e.g. Given one training example (x, y).



Forward propagation:

$$a^{(1)} = x$$
$$z^{(2)} = \Theta^{(1)} a^{(1)}$$
$$a^{(2)} = g(z^{(2)}) \quad (\text{add } a_0^{(2)})$$
$$z^{(3)} = \Theta^{(2)} a^{(2)}$$
$$a^{(3)} = g(z^{(3)}) \quad (\text{add } a_0^{(3)})$$
$$z^{(4)} = \Theta^{(3)} a^{(3)}$$
$$a^{(4)} = g(z^{(4)}) = h_\Theta(x)$$

Backpropagation:

Calculate $\delta_j^{(l)}$, the "error" of node j in layer l.

For each output unit (layer $L = 4$):

$$\delta_j^{(4)} = a_j^{(4)} - y_j$$

$(h_\Theta(x))_j$

$$\delta^{(3)} = (\Theta^{(3)})^T \delta^{(4)} .* g'(z^{(3)}) \qquad a^{(3)} .* (1 - a^{(3)})$$
$$\delta^{(2)} = (\Theta^{(2)})^T \delta^{(3)} .* g'(z^{(2)}) \qquad a^{(2)} .* (1 - a^{(2)})$$

Backpropagation Algorithm:

Set $\Delta_{ij}^{(l)} = 0$ for all i, j, l.

For $i = 1$ to m (# training examples),

Set all $a^{(1)} = x^{(i)}$.

Perform forward prop to compute $a^{(l)}$ for $l = 2, 3, \ldots L$.

Using $y^{(i)}$, compute $\delta^{(L)} = a^{(L)} - y^{(i)}$.

Compute $\delta^{(L-1)}, \delta^{(L-2)} \ldots \delta^{(2)}$ ~~$\delta^{(1)}$~~

$\Delta_{ij}^{(l)} := \Delta_{ij}^{(l)} + a_j^{(l)} \delta_i^{(l+1)}$

$D_{ij}^{(l)} := \frac{1}{m} \Delta_{ij}^{(l)} + \lambda \theta_{ij}^{(l)}$ if $j \neq 0$

$D_{ij}^{(l)} := \frac{1}{m} \Delta_{ij}^{(l)}$ if $j = 0$

You'll find that $\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) = D_{ij}^{(l)}$.

✡Calculated once FP, BP done for all examples.

✡Do FP for one example, then BP, then FP for the next example.