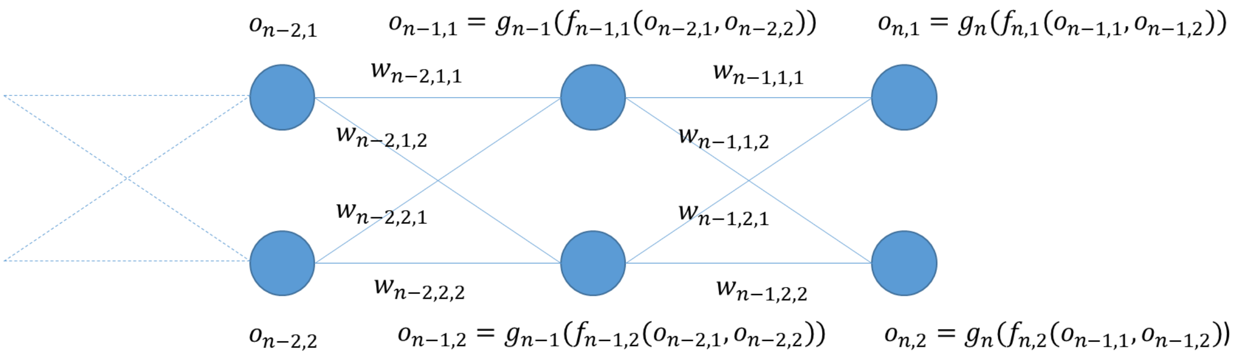
Simple Illustration of Programmable Backpropagation

Lei Mao
Department of Computer Science
University of Chicago
5/17/2017

Introduction

Suppose this is our neural network. I am only showing the last three layers of it.


$o_{n-2,1}$
$o_{n-1,1} = g_{n-1}(f_{n-1,1}(o_{n-2,1}, o_{n-2,2}))$
$o_{n,1} = g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))$
$w_{n-2,1,1}$
$w_{n-1,1,1}$
$w_{n-2,1,2}$
$w_{n-1,1,2}$
$w_{n-2,2,1}$
$w_{n-1,2,1}$
$w_{n-2,2,2}$
$w_{n-1,2,2}$
$o_{n-2,2}$
$o_{n-1,2} = g_{n-1}(f_{n-1,2}(o_{n-2,1}, o_{n-2,2}))$
$o_{n,2} = g_n(f_{n,2}(o_{n-1,1}, o_{n-1,2}))$

$f_{n,1}(o_{n-1,1}, o_{n-1,2}) = w_{n-1,1,1}o_{n-1,1} + w_{n-1,2,1}o_{n-1,2}$ where w is the weights connecting different neurons and o is the output from the neuron.

$o_{n,1} = g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))$ where function $g_n(x)$ is the activation function at layer n.

Suppose the target value for $o_{n,1}$ on the last layer is t_1, we could compute the error δ_1 using a certain loss function:

$$\delta_1 = loss(o_{n,1}, t_1)$$

Then we could backpropagate this error to each weights at each layer. The main idea of back propagation is multivariable chain rule. We will calculate the derivatives at the last layer n, and used the derivatives calculated in the layer n to calculate derivatives in the layer $n-1$, and so on. The backpropagated errors from one layer to another are colored using different colors. We will see that this backpropagation follows certain rules and is totally programmable.

Backpropagation

Calculate error derivatives

$$\frac{\partial \delta_1}{\partial o_{n,1}} = \frac{\partial loss(o_{n,1}, t_1)}{\partial o_{n,1}}$$

Layer n

Calculate (you don't even have to do some of them)

$$\frac{\partial g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))}{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}$$

$$\frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial w_{n-1,1,1}} = o_{n-1,1}$$

$$\frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial w_{n-1,2,1}} = o_{n-1,2}$$

$$\frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial o_{n-1,1}} = w_{n-1,1,1}$$

$$\frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial o_{n-1,2}} = w_{n-1,2,1}$$

Further calculate

$$\frac{\partial o_{n,1}}{\partial w_{n-1,1,1}} = \frac{\partial g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))}{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})} \frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial w_{n-1,1,1}}$$

$$\frac{\partial o_{n,1}}{\partial w_{n-1,2,1}} = \frac{\partial g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))}{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})} \frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial w_{n-1,2,1}}$$

$$\frac{\partial o_{n,1}}{\partial o_{n-1,1}} = \frac{\partial g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))}{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})} \frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial o_{n-1,1}}$$

$$\frac{\partial o_{n,1}}{\partial o_{n-1,2}} = \frac{\partial g_n(f_{n,1}(o_{n-1,1}, o_{n-1,2}))}{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})} \frac{\partial f_{n,1}(o_{n-1,1}, o_{n-1,2})}{\partial o_{n-1,2}}$$

Weight derivatives in layer $\boldsymbol{n}$

$$\frac{\partial \delta_1}{\partial w_{n-1,1,1}} = \frac{\partial \delta_1}{\partial o_{n,1}} \frac{\partial o_{n,1}}{\partial w_{n-1,1,1}}$$

$$\frac{\partial \delta_1}{\partial w_{n-1,2,1}} = \frac{\partial \delta_1}{\partial o_{n,1}} \frac{\partial o_{n,1}}{\partial w_{n-1,2,1}}$$

Derivatives used in layer $\boldsymbol{n-1}$

$$\frac{\partial \delta_1}{\partial o_{n-1,1}} = \frac{\partial \delta_1}{\partial o_{n,1}} \frac{\partial o_{n,1}}{\partial o_{n-1,1}}$$

$$\frac{\partial \delta_1}{\partial o_{n-1,2}} = \frac{\partial \delta_1}{\partial o_{n,1}} \frac{\partial o_{n,1}}{\partial o_{n-1,2}}$$

Layer $\boldsymbol{n-1}$

Calculate (you don’t even have to do some of them)

$$\frac{\partial g_{n-1}(f_{n-1,1}(o_{n-2,1}, o_{n-2,2}))}{\partial f_{n-1,1}(o_{n-2,1}, o_{n-2,2})}$$

$$\frac{\partial g_{n-1}(f_{n-1,2}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}$$

$$\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,1,1}} = o_{n-2,1}$$

$$\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,1,2}} = o_{n-2,1}$$

$$\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,2,1}} = o_{n-2,2}$$

$$\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,2,2}} = o_{n-2,2}$$

$$\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,1}} = w_{n-2,1,1}$$

$$\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,1}} = w_{n-2,1,2}$$

$$\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,2}} = w_{n-2,2,1}$$

$$\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,2}} = w_{n-2,2,2}$$

Further calculate

$$\frac{\partial o_{n-1,1}}{\partial w_{n-2,1,1}} = \frac{\partial g_{n-1}(f_{n-1,1}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,1,1}}$$

$$\frac{\partial o_{n-1,2}}{\partial w_{n-2,1,2}} = \frac{\partial g_{n-1}(f_{n-1,2}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,1,1}}$$

$$\frac{\partial o_{n-1,1}}{\partial w_{n-2,2,1}} = \frac{\partial g_{n-1}(f_{n-1,1}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,2,1}}$$

$$\frac{\partial o_{n-1,2}}{\partial w_{n-2,2,2}} = \frac{\partial g_{n-1}(f_{n-1,2}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial w_{n-2,2,2}}$$

$$\frac{\partial o_{n-1,1}}{\partial o_{n-2,1}} = \frac{\partial g_{n-1}(f_{n-1,1}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,1}}$$

$$\frac{\partial o_{n-1,1}}{\partial o_{n-2,2}} = \frac{\partial g_{n-1}(f_{n-1,1}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,1}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,2}}$$

$$\frac{\partial o_{n-1,2}}{\partial o_{n-2,1}} = \frac{\partial g_{n-1}(f_{n-1,2}(o_{n-2,1},o_{n-2,2}))}{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}\frac{\partial f_{n-1,2}(o_{n-2,1},o_{n-2,2})}{\partial o_{n-2,1}}$$

$$\frac{\partial o_{n-1,2}}{\partial o_{n-2,2}} = \frac{\partial g_{n-1}(f_{n-1,2}(o_{n-2,1}, o_{n-2,2}))}{\partial f_{n-1,2}(o_{n-2,1}, o_{n-2,2})} \frac{\partial f_{n-1,2}(o_{n-2,1}, o_{n-2,2})}{\partial o_{n-2,2}}$$

Weight derivatives in layer $n - 1$

$$\frac{\partial \delta_1}{\partial w_{n-2,1,1}} = \frac{\partial \delta_1}{\partial o_{n-1,1}} \frac{\partial o_{n-1,1}}{\partial w_{n-2,1,1}}$$

$$\frac{\partial \delta_1}{\partial w_{n-2,1,2}} = \frac{\partial \delta_1}{\partial o_{n-1,2}} \frac{\partial o_{n-1,2}}{\partial w_{n-2,1,2}}$$

$$\frac{\partial \delta_1}{\partial w_{n-2,2,1}} = \frac{\partial \delta_1}{\partial o_{n-1,1}} \frac{\partial o_{n-1,1}}{\partial w_{n-2,2,1}}$$

$$\frac{\partial \delta_1}{\partial w_{n-2,2,2}} = \frac{\partial \delta_1}{\partial o_{n-1,2}} \frac{\partial o_{n-1,2}}{\partial w_{n-2,2,2}}$$

Derivatives used in layer $n - 2$ (multivariable chain rule)

$$\frac{\partial \delta_1}{\partial o_{n-2,1}} = \frac{\partial \delta}{\partial o_{n-1,1}} \frac{\partial o_{n-1,1}}{\partial o_{n-2,1}} + \frac{\partial \delta}{\partial o_{n-1,2}} \frac{\partial o_{n-1,2}}{\partial o_{n-2,1}}$$

$$\frac{\partial \delta_1}{\partial o_{n-2,2}} = \frac{\partial \delta}{\partial o_{n-1,1}} \frac{\partial o_{n-1,1}}{\partial o_{n-2,2}} + \frac{\partial \delta}{\partial o_{n-1,2}} \frac{\partial o_{n-1,2}}{\partial o_{n-2,2}}$$

Layer $n - 2$

Use the same way to calculate derivatives…

Weight derivatives in layer $n - 2$

$$\frac{\partial \delta_1}{\partial w_{n-3,1,1}} = \frac{\partial \delta_1}{\partial o_{n-2,1}} \frac{\partial o_{n-2,1}}{\partial w_{n-3,1,1}}$$

$$\frac{\partial \delta_1}{\partial w_{n-3,1,2}} = \frac{\partial \delta_1}{\partial o_{n-2,2}} \frac{\partial o_{n-2,2}}{\partial w_{n-3,1,2}}$$

$$\frac{\partial \delta_1}{\partial w_{n-3,2,1}} = \frac{\partial \delta_1}{\partial o_{n-2,1}} \frac{\partial o_{n-2,1}}{\partial w_{n-3,2,1}}$$

$$\frac{\partial \delta_1}{\partial w_{n-3,2,2}} = \frac{\partial \delta_1}{\partial o_{n-2,2}} \frac{\partial o_{n-2,2}}{\partial w_{n-3,2,2}}$$

Derivatives used in layer $n - 3$ (multivariable chain rule)

$$\frac{\partial \delta_1}{\partial o_{n-3,1}} = \frac{\partial \delta}{\partial o_{n-2,1}} \frac{\partial o_{n-2,1}}{\partial o_{n-3,1}} + \frac{\partial \delta}{\partial o_{n-2,2}} \frac{\partial o_{n-2,2}}{\partial o_{n-3,1}}$$

$$\frac{\partial \delta_1}{\partial o_{n-3,2}} = \frac{\partial \delta}{\partial o_{n-2,1}} \frac{\partial o_{n-2,1}}{\partial o_{n-3,2}} + \frac{\partial \delta}{\partial o_{n-2,2}} \frac{\partial o_{n-2,2}}{\partial o_{n-3,2}}$$

Summary

We can see from the above derivation, it could be a simple iterative computer program.

Basically, what we are doing in the backpropagation program is that, at each layer n, we calculate the derivatives of activation function at each neuron, with a combination of the weights and outputs from the previous layer $n-1$, and the propagated errors of layer n which was already calculated at layer $n+1$.

I am personally not interested in writing a program of doing backpropagations or autograds, since there are already tons of them. But this material unveil the programmable nature of backpropagation while most of the other materials I read previous did not have any concept of programming when they were introducing backpropagation. They usually just provide some simple chain rule examples without even providing a real neuron network. So the reader could only have some idea that the backpropagation is about the chain rules. But for programmers, we still do not quite understand such complicated derivations could be done in simple computer programs. The backpropagation program then become a black box even we are using it every day.