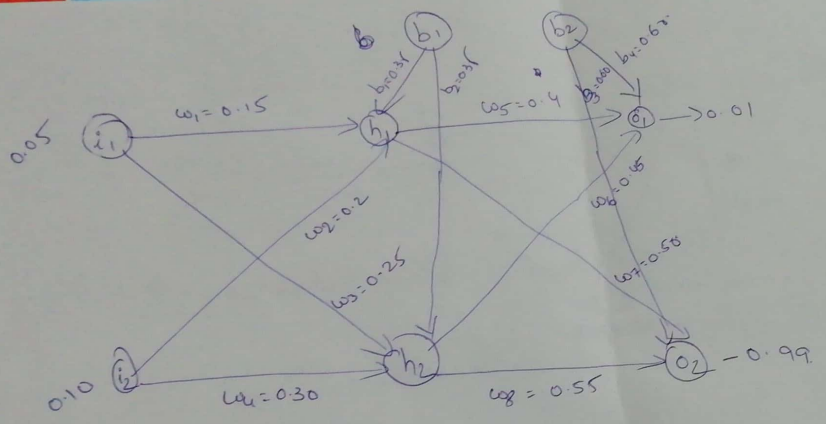
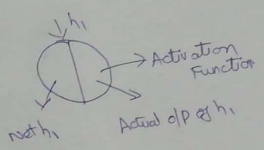




1st Forward Propogation

$net h_1 = \omega_1 i_1 + \omega_2 x i_2 + b_1 = 0.15 \times 0.05 + 0.2 \times 0.1 + 0.35$

$net h_1 = 0.3775$

$out h_1 = \text{Sigmoid function} = \frac{1}{1+e^{-0.3775}} =$

$out h_1 = 0.593269992$

Similarly,

$out h_2 = 0.596884370$

$net o_1 = \omega_5 \times out h_1 + \omega_6 \times out h_2$

$= 0.4 \times 0.593269992 + 0.45 \times 0.596884370$

$output\ o_1 = 1.105967$

$output\ o_1 = \frac{1}{1-e^{1.105967}} = 0.751365047$

Similarly

$output\ o_2 = 0.772928465$

$w_5 = w_5 - \eta$

Backward Propagation

$E_{o1} = \frac{1}{2}(\text{target } o1 - \text{output } o_1)^2$

$= \frac{1}{2}(0.01 - 0.7513)^2$

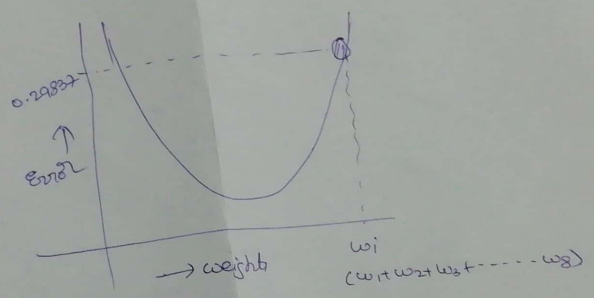
$E_{o1} = 0.274811083$

Similarly

$E_{o2} = 0.023560026$

Total Error $E_{total} = E_{o1} + E_{o2}$

$E_{total} = 0.298371109$



$\frac{1}{2}[\text{target } o_2 -$

$+ G_{6\times a}$

New Fun Filled!

$$w_5 = w_5 - \left(\frac{\partial E}{\partial w_5}\right) \times \eta$$ (learning rate)

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_1} \times \frac{\partial out_{01}}{\partial net_{01}} \times \frac{\partial net_{01}}{\partial w_5}$$

$$\frac{\partial E_{total}}{\partial out_1} = \frac{\partial}{\partial out} \left[\frac{1}{2}(target_{01} - out_{01})^2 + \frac{1}{2}(target_{02} - out_{02})^2 \right]$$

Constant = 0.

$$= \frac{1}{2} \times 2 \, (target_{01} - out_{01})^{2-1} - \frac{\partial out_1}{\partial out}$$

$$= (target_{01} - out_{01}) \times -1$$

$$- out_{01} - target_{01} = 0.7513 - 0.01$$

$$= 0.7413$$

$$\frac{\partial E_{total}}{\partial out_{01}} = 0.7413$$

$$\frac{\partial out_{01}}{\partial net_{01}} = \frac{\partial}{\partial net_{01}} out_{01}$$

$$= \frac{\partial}{\partial net_{01}} \frac{1}{1+e^{-net_{01}}} =$$

$$= out_{01}(1 - out_{01}) = 0.75(1-0.75) = 0.1868156 02$$

$$E_{total} = \frac{1}{2}[target_{01} - out_{01}]^2 + \frac{1}{2}[target_{02} - out_{02}]^2$$

$$=$$

$$\frac{\partial net_{01}}{\partial w_5} = w_5 \times out_{h_1} + w_6 \times out_{h_2} + 1 \times b_2$$

$$= out_{h_1}.$$

$$\frac{\partial net_{01}}{\partial w_5} = 0.593269992$$

$$w_5^* = w_5 - \left(\frac{\partial E_{total}}{\partial out} \times \frac{\partial out_{01}}{\partial net_{01}} \times \frac{\partial net_{01}}{\partial w_5}\right) \times \eta$$

$$= 0.4 - (0.7413 \times 0.186815602 \times 0.593269992) \times 0.5$$

$$w_5^+ = 0.35892009$$

$(1/1+e^{-z})$ $\frac{e^z - e^{-z}}{e^z + e^{-z}}$ $\frac{d}{dz}$ → $1 - \tanh^2(z)$

→ activation function (Sigmoid, Tanh $0 \le \frac{\tanh}{\partial z} \le 1$, $-1 \le 1$, ReLU, $\begin{cases} 0 & y \le 0 \\ z & o/w \end{cases}$

Noisyrelu $\max(0, x+Y)$

leaky relu $\begin{cases} x & y\ x>0 \\ 0.01 & o/w \end{cases}$

Softplus $\begin{cases} 0 & y\ z<0 \\ 1 & y\ z>0 \end{cases}$

→ memorisation

→ chain rule (Training MLP)

→ Back Propagation

→ Vanishing gradient

→ Hyper Parameter

→ Dropout (Random subset of features)

→ Regularization

→ Batch normalization

→ weights initialization. (uniform initialization (Sigmoid) $\left[\frac{-1}{\sqrt{fanin}}, \frac{1}{\sqrt{fanout}}\right]$, Xavier/Glorot init (Sigmoid) $\left[\frac{-\sqrt{6}}{\sqrt{fanin+fanout}}, \frac{+\sqrt{6}}{\sqrt{fanin+fanout}}\right]$,

He-init
normal $= \sqrt{\frac{2}{fanin}}$
uniform $= \left[\frac{-\sqrt{6}}{\sqrt{fanin}}, \frac{+\sqrt{6}}{\sqrt{fanout}}\right]$

→ optimizers

↳ SGD, AdaGrad, Adadelta, Adam

$w_t = w_{t-1} - \eta' t g_t$

$\eta' t = \frac{\eta \to 0.01}{\sqrt{\alpha_{t-1} + \epsilon}}$

$\alpha_{t-1} = \sum_{i=1}^{t} g_i^2 \to \left(\frac{\partial L}{\partial w}\right)_{(w t-1)}$

$\eta' t = \frac{\eta}{\sqrt{\text{Edat}_{t-1} + \epsilon}}$

$\text{Edat}_{t-1} = \gamma\, \text{Edat}_{t-2} + (1-\gamma) g_{t-1}^2$

MB-SGD + Momentum NAG → Adagrad → Adadelta → Adam

Small/shallow NN

work in most cases (Stochastic)

Sparse

favorite

→ loss (from keras import losses)

↳ mean_squared_error, mean_squared_logarithmic, mean_Absolute_Error_loss

↳ hinge_loss, squared_hinge

↳ Categorical_Crossentropy

↳ binary_Cross

$(1/1+e^{-z})$ $\frac{e^z - e^{-z}}{e^z + e^{-z}}$ $\frac{dy}{dz} \rightarrow 1 - \tanh^2(z)$

$0 \leq \frac{\tanh}{dz} \leq 1$, $-1 \leq 1$

→ activation function (Sigmoid, Tanh, ReLU, $\begin{cases} 0 & y\ z \leq 0 \\ z & o/w \end{cases}$

Noisy relu $\max(0, x+Y)$

leaky relu $\begin{cases} x & y\ x>0 \\ 0.01 & o/w \end{cases}$

soft plus $\begin{cases} 0 & y\ z<0 \\ 1 & y\ z>0 \end{cases}$

→ memorisation

→ chain rule (Training MLP)

→ Back Propagation

→ Vanishing gradient

→ Hyper Parameter

→ Drop out (Random subset of features)

→ Regularization.

→ Batch normalization

→ weights initialization. (Uniform initialization (Sigmoid) $\left[\frac{-1}{\sqrt{fanin}}, \frac{1}{\sqrt{fanout}}\right]$, Xavier/Glorot init (Sigmoid) $\left[\frac{-\sqrt{6}}{\sqrt{fanin+fanout}}, \frac{+\sqrt{6}}{\sqrt{fanin+fanout}}\right]$, He-init normal $= \sqrt{\frac{2}{fanin}}$, uniform $= \left[\frac{-\sqrt{6}}{\sqrt{fanin}}, \frac{+\sqrt{6}}{\sqrt{fanout}}\right]$

→ optimizers

↳ SGD, AdaGrad, Adadelta, Adam

$w_t = w_{t-1} - \eta' t g_t$

$\eta' t = \frac{\eta \to 0.01}{\sqrt{\alpha_{t-1} + \epsilon}}$

$\alpha_{t-1} = \sum_{i=1}^{t} g_i^2 \rightarrow \left(\frac{\partial L}{\partial w}\right)_{(w t-1)}$

$\eta' t = \frac{\eta}{\sqrt{edat_{t-1} + \epsilon}}$

$edat_{t-1} = \gamma\, edat_{t-2} + (1-\gamma) g_{t-1}^2$

MB-SGD + Momentum NAG → Adagrad → Adadelta → Adam

Small/shallow NN

Momentum NAG ↓ work in most cases (slower)

Adagrad → Sparse

Adam → favourite

→ loss (from keras import losses)

↳ mean_square_error, mean square logarithmic, mean Absolute Error loss

↳ hinge loss, square hinge

↳ Categorical_Crossent

↳ binary_Cross