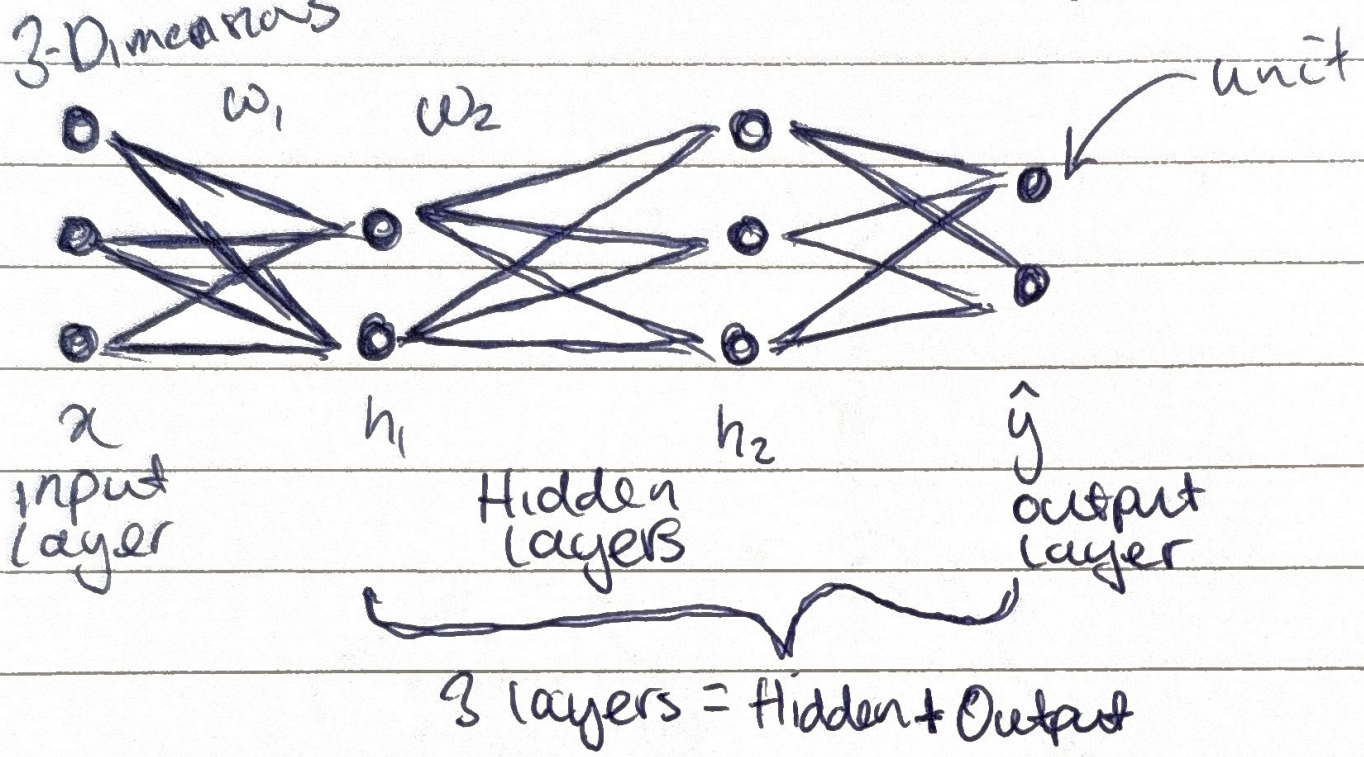
Multi-layer Perceptron

Smallest MLP = 2 layers



- each unit connects to each unit in the next
- No intra layer connections

feed forward example:

input $\xrightarrow{\text{linear map}}$ Hidden $\longrightarrow$ output

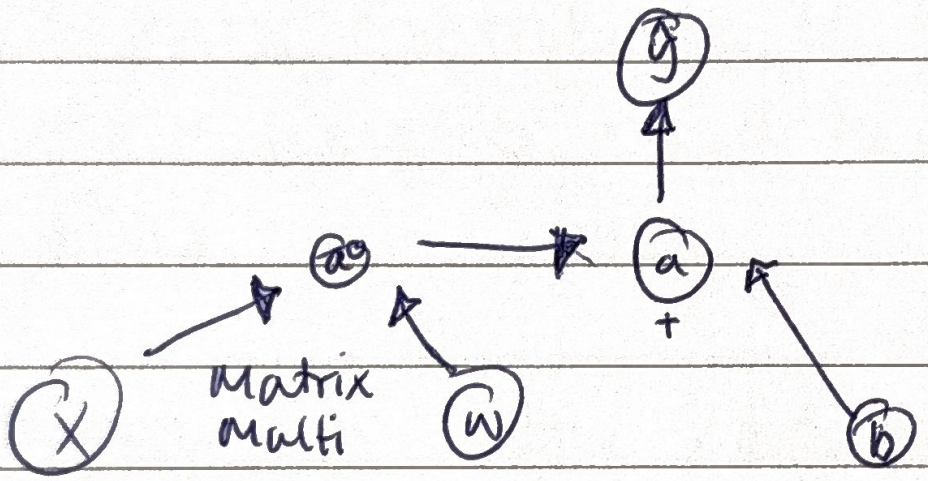
$h(x) = w_1 x$

Computational Graphs

Nodes = variabls (input, output, Parametes

Edges = Mapping

$$\hat{y} = \text{relu}(x w + b)$$



Backpropagation

backpropigating the loss gradients from the output layer to all of the layers

All layers contribute to the loss so their weights need to be updated in SGD

Backprop is the process by which the loss gradient computed @ the output is propagated backward to the inner layers

Output layer is a function of a function so we need chain rule for Diff

because we have multiple dimensions (i.e. units) we need to compute partial Derivs ∂x instead of dx

Backpropagation equation Derivation

$$\frac{\partial L}{\partial w_{ij}} = \frac{\partial L}{\partial Z_j} \cdot \frac{\partial Z_j}{\partial w_{ji}}$$ (chain rule)

where ↖ loss gradient for the weight ji connecting unit i in layer T-1 to unit j in layer T

$Z_j = \sum_i a_i w_{ji}$ = linear output model for unit j

a_i = the input for unit j = activation output for unit i $g(z_i)$

↳ Denote $\frac{\partial L}{\partial Z_j}$ as ∇_j

$$= \frac{\partial L}{\partial w_{ji}} = \nabla_j \cdot \frac{\partial Z_j}{\partial w_{ij}}$$

Sub $Z_j = \sum_i a_i w_{ji}$ & perform DIFF

$$\frac{\partial L}{\partial w_{ji}} = \nabla_j \cdot a_i$$

Sub $a_i = g(z_i)$ $= \nabla j \cdot g(z_i)$

Solving ∇_j

j represents a layer & a layer has n number of units

u will represent a unit in the layer

$$\Rightarrow \nabla_u = \frac{\partial L}{\partial z_u}$$

apply the chain rule

$$\nabla_u = \sum \frac{\partial L}{\partial z_k} \cdot \frac{\partial z_k}{\partial z_u}$$

denoting $\frac{\partial L}{\partial z_k}$ as ∇_k : $\nabla_u = \sum \nabla_k \cdot \frac{\partial z_k}{\partial z_u}$

recall $z_k = \sum a_j \cdot w_{kj}$ & then $a_j = g(z_j)$

more steps in slides

Unit = Dimension

final form

$$\frac{\partial L}{\partial w_{ji}} = \left(g'(z_j) \sum \nabla_k \cdot w_{kj} \right) \cdot g(z_i)$$

Derivative of activation for unit j in layer T

activation for unit i in layer T-1

loss gradient for unit K in layer t+1

Output layer has to t+1 layer $= \frac{\partial L}{\partial \hat{y}} \cdot g(z_i)$

Input layer has no t-1 activation layer

$= (\sim\sim) \cdot \underline{x_i}$