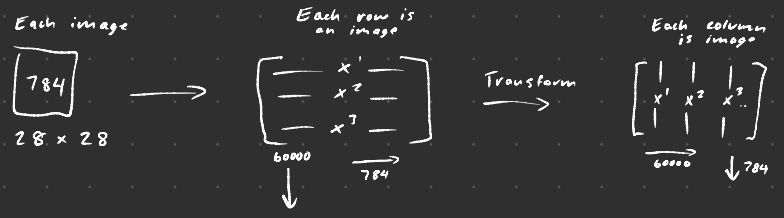
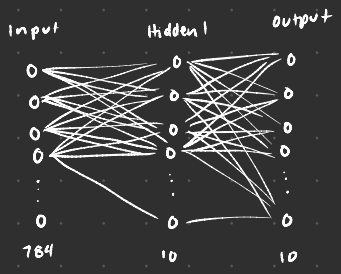
Data Format



Network Structure



Forward Propagation → starts with input image, ends with prediction output

Input layer

$A^0 = X \quad (784 \times m)$

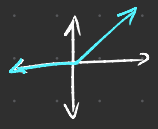
Hidden — weight / dot product — bias

$Z^1 = W^1 A^0 + b^1$

$(10 \times m) \quad (10 \times 784) \quad (784 \times m) \quad (10 \times 1 \Rightarrow 10 \times m)$

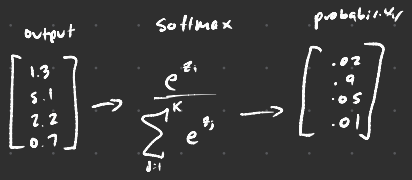
Activation Function

$A^1 = g(z^1) = ReLU(z^1)$ ReLU = Rectified linear unit $= \begin{cases} x \text{ if } x > 0 \\ 0 \text{ if } x \leq 0 \end{cases}$



$Z^2 = W^2 A^1 + b^2$

$A^2 = \text{softmax}(Z^2)$



Backward Propagation → update weights & biases for next forward prop

Starts with prediction, calculate how far prediction deviated from label

Find out how much W & b contributed from that deviation, and adjust

How much was prediction off by?

Error of Hidden layer

$dz^2 = A^2 - Y$ (Prediction − label)

How did W & b contribute to loss

$dw^2 = \frac{1}{m} dz^2 A^{1T}$ derivative of loss function

$db^2 = \frac{1}{m} \sum dz^1$

$dz^1 = w^{2T} dz^2 .* g'(z^1)$ derivative of activation function

$dw^1 = \frac{1}{m} dz^1 x^T$

$db^2 = \frac{1}{m} \sum dz^1$

Update W & b — α = learning rate multiplier

$W^1 = W^1 - \alpha\, dw^1$

$b^1 = b^1 - \alpha\, db^1$

$W^2 = W^2 - \alpha\, dw^2$

$b^2 = b^2 - \alpha\, db^2$

Back to forward Prop!

Rinse & repeat ALOT