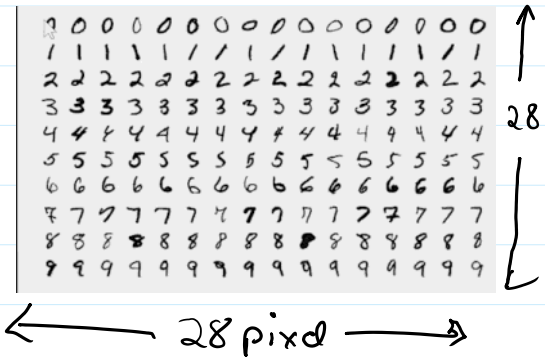
Building a neural network FROM SCRATCH

06 May 2024 18:08

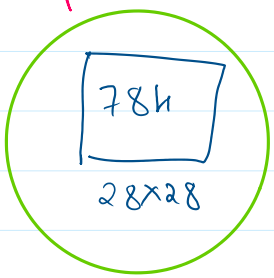
→ mnist data set



X m

is the volume of dataset

⇒ implementation



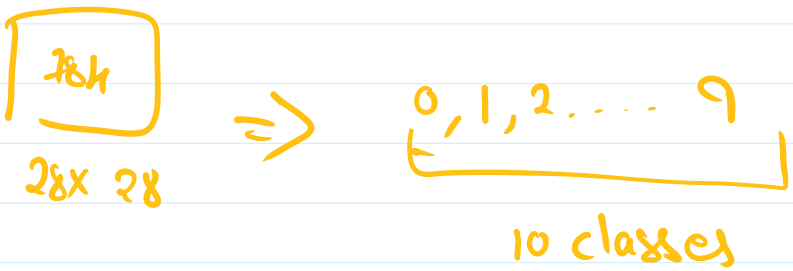
time's m ; traning images

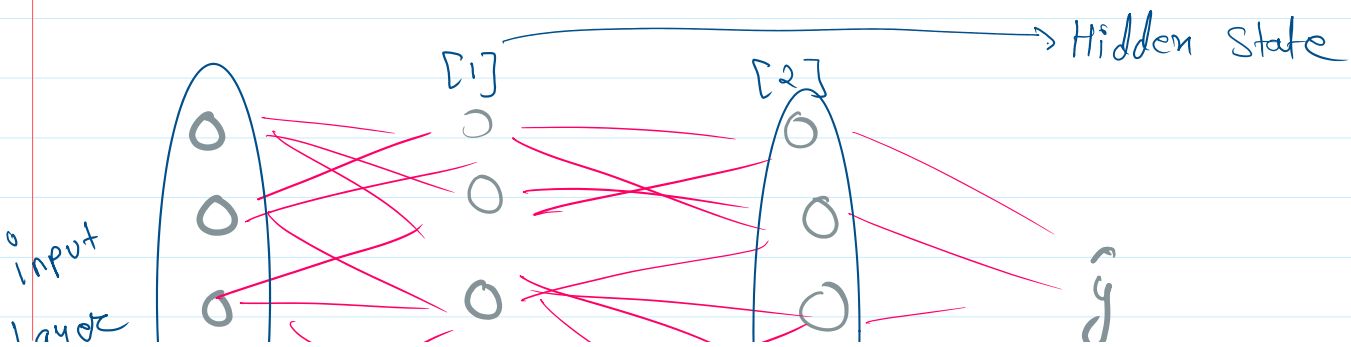
x*m is the traning dataset

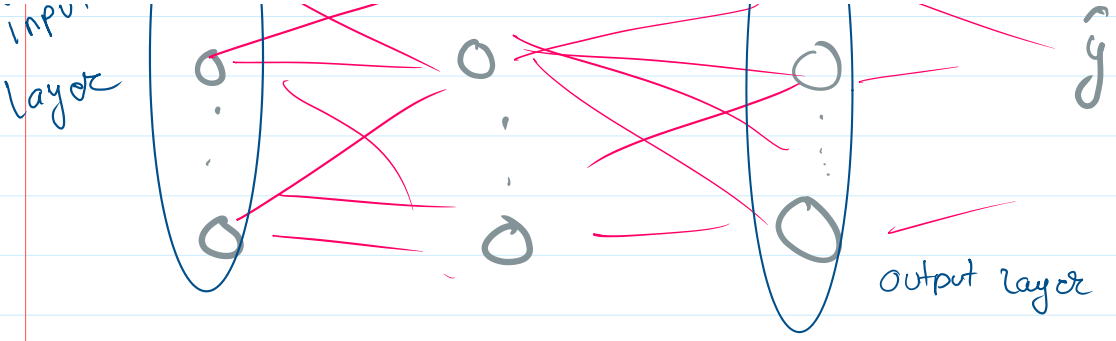
x

$$x = \begin{bmatrix} - & x^1 & - \\ - & x^2 & - \\ - & x^3 & - \\ - & x^m & - \end{bmatrix}^T = \begin{bmatrix} | & | & | & | & & | \\ x^1 & x^2 & x^3 & x^4 & \cdots & x^m \\ | & | & | & | & & | \end{bmatrix}$$

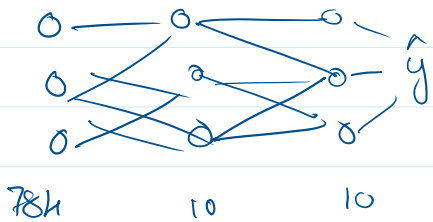
total dataset







(1) Forward Propogation



$$A^{[0]} = X \quad \{784 \times m\}$$

Unactivated 1st layer

$$Z^{[1]} = W^{[1]} \cdot A^{[0]} + b^{[1]}$$

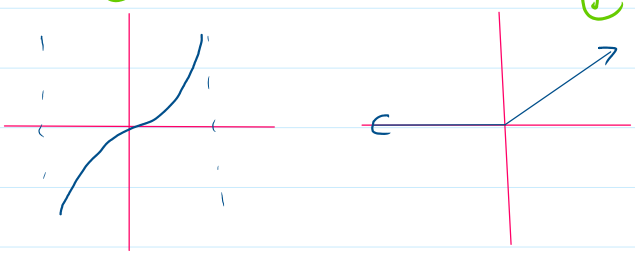
$10 \times m \quad 10 \times 784 \quad 784 \times m \quad 10 \times 1 \to 10 \times m$

{ w → weight, b → bias }

Activation Function

$$A^{[1]} = g(Z^{[1]}) = ReLU(Z^{[1]})$$

1) without it, it would be a fancy linear Regression



tan(h)

rectified linear

$$ReLU = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}$$

Unactivated 2nd layer

$$Z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$$

$10 \times m \quad 10 \times 10 \quad 10 \times m \quad 10 \times 1 \Rightarrow 10 \times m$

Activation function

$$A^{[2]} = \text{Softmax}\left(Z^{[2]}\right)$$

Output layer → Softmax activation function → Probabilities

$$\frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$

(2) Backward Propogation

$$dZ^{[2]} = A^{[2]} - Y$$

$10 \times m \quad 10 \times m \quad 10 \times m$

{ $dz^{[2]}$ ← error of 2nd layer ; y → one Hotencode }

$$dW^{[2]} = \frac{1}{m} dZ^{[2]} A^{[1]T}$$

$10 \times 10 \quad 10 \times m \quad m \times 10$

{ $dw^{[2]}$ = derivative of Cost function }

$$db^{[2]} = \frac{1}{m} \Sigma dZ^{[2]}$$

$10 \times 1 \quad 10 \times 1$

{ $db^{[2]}$ = avg of the absolute error }

error for 1st layer

$$dZ^{[1]} = W^{[2]T} dZ^{[2]} * g'(Z^{[1]})$$

$10 \times m \quad 10 \times 10 \quad 10 \times m \quad 10 \times m$

→ derivative of the activation function (derivative of ReLU)

Contribution of $W^{[1]}$ to the error

$$dW^{[1]} = \frac{1}{m} dZ^{[1]} X^T$$

$10 \times 784 \quad 10 \times m \quad m \times 784$

of $w^{[1]}$ — m

to the error 10×784 $10 \times m$ $m \times 784$

Contribution of $b^{[1]}$ to the error $\quad db^{[1]} = \frac{1}{m} \sum dz^{[1]}$

10×1 10×1

(3) Parameters updation

learning rate

$$w^{[1]} := w^{[1]} - \alpha\, dw^{[1]}$$
$$b^{[1]} := b^{[1]} - \alpha\, db^{[1]}$$
$$w^{[2]} := w^{[2]} - \alpha\, dw^{[2]}$$
$$b^{[1]} := b^{[2]} - \alpha\, db^{[2]}$$