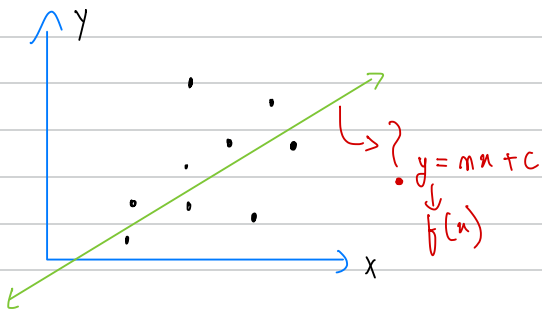
Regression

x	y
10	100
20	190
30	370
40	420
⋮	⋮
	(n)



Squared Error Cost Function (F)

$$F = \frac{1}{2n} \sum_{i=1}^{n} \left(f(x^i) - y^i \right)^2$$

$f(x^i) \rightarrow mx + c$

- minimize F
- Gradient Descent

Set $(m, c) = (0, 0)$

$$m = m - \alpha * \frac{\partial F(m,c)}{\partial m}, \quad c = c - \alpha * \frac{\partial F(m,c)}{\partial c}$$

MNIST

① Problem Definition

- 28x28 images of digits with labels 0-9
- Train a model to classify images into 0-9

② Data Collection

- Loading data from keras
  - It has MNIST available in it.

③ Data Pre-Processing

- What is the data like (dimensions, content etc.)

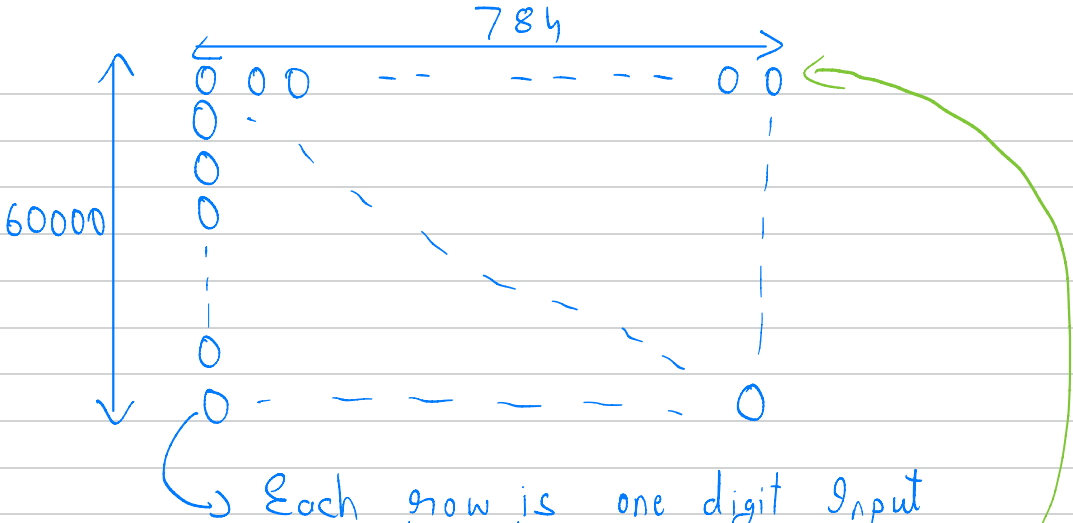
① -> Normalization

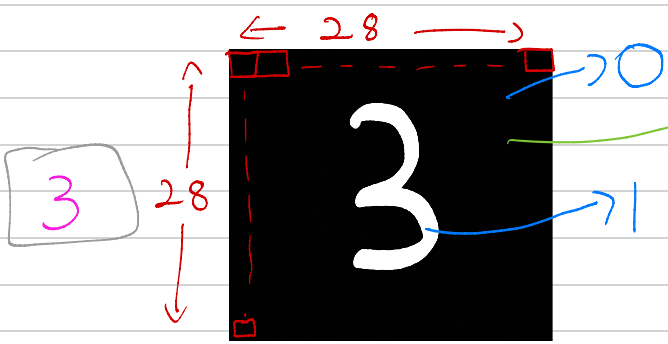
- Improves Speed (lesser processing time)
- Prevents large gradient values
- Avoids giving power to features with large values.

② Reshaping the data according to the model.

It's (60000, 28, 28) — 28*28
(z, y, x)

Reshape to (60000, 784)
(y, x)

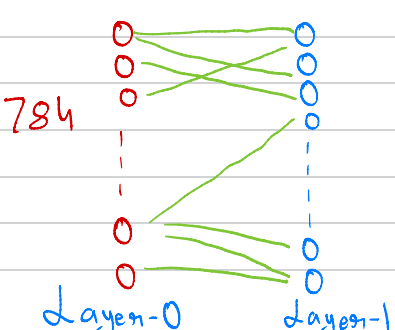




Augmentation
Standardization
Noise Reduction

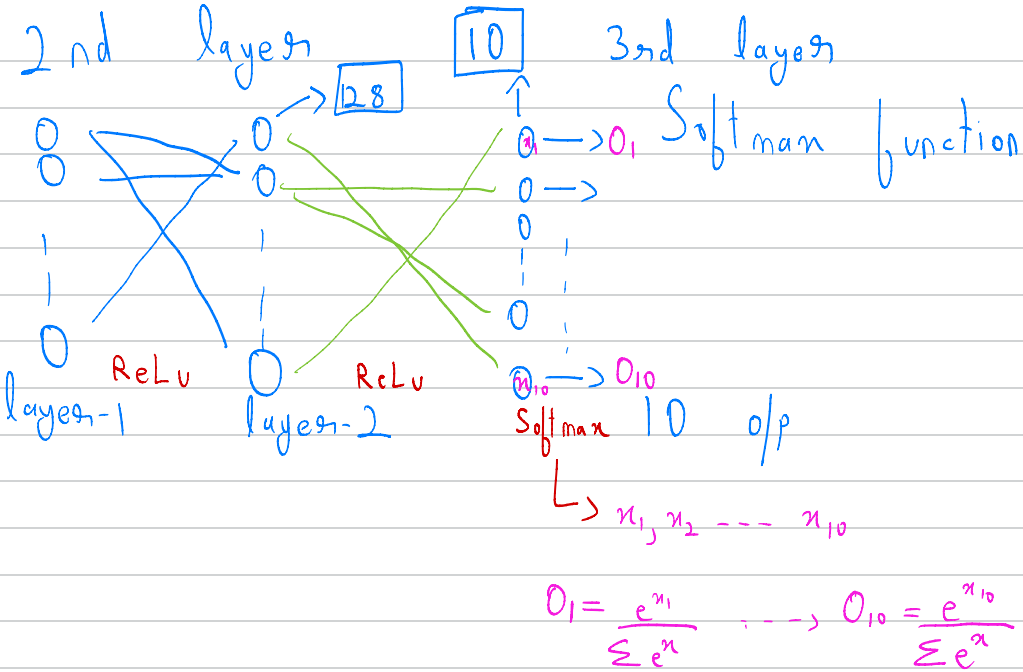
⑤ Choosing a model

Neural Network



1st layer

$f = \max(0, x)$



Adam -> Changes Learning Rate according to requirement
Loss -> A Loss function used in Multi - Class classification

(5) Train Model

(6) Evaluate