deeplearning.ai

Basics of Neural Network Programming

Binary Classification

Binary Classification

64

64

1 (cat) vs 0 (non cat)

y = output variable

Blue

Green

Red

		255	134	93	22
	255	134	202	22	2
255	231	42	22	4	30
123	94	83	2	192	124
34	44	187	92	34	142
34	76	232	124	94	
67	83	194	202		

64

64

$$X = \begin{bmatrix} 255 & 255 & 255 \\ 231 & 134 & 134 \\ 42 & \vdots & \vdots \\ \vdots & & \\ 202 & 94 & 142 \end{bmatrix} \Rightarrow 64 \times 64 \times 3 = n_x \text{ or } n$$

OR

$$X = \begin{bmatrix} 255 \\ 231 \\ \vdots \\ 202 \\ 255 \\ 134 \\ \vdots \\ 94 \\ \vdots \\ 142 \end{bmatrix}$$

Andrew Ng

Notation

$(x, y) \longrightarrow x \in \mathbb{R}^{n_x}, \quad y \in \{0, 1\}$

m training examples: $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}) \dots (x^{(m)}, y^{(m)})\}$

$m_{train}, \quad m_{test}$

$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \dots & x^{(m)} \\ | & | & & | \end{bmatrix} \quad \uparrow n_x \downarrow$$

$\longleftarrow m \longrightarrow$

$X \in \mathbb{R}^{n_x \times m}$

→ note we are rep examples with "m columns" ⇒ #training examples are rep in the columns, not rows! (why? This will work better in NN)

$X.shape = (n_x, m)$

$Y = [y^{(1)}, y^{(2)} \dots y^{(m)}]$

$\Rightarrow Y \in \mathbb{R}^{1 \times m}$

Again note # examples "m" comes in columns

$\Rightarrow Y.shape(1, m)$