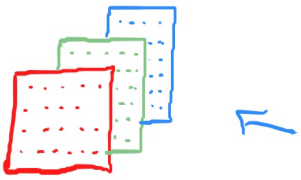
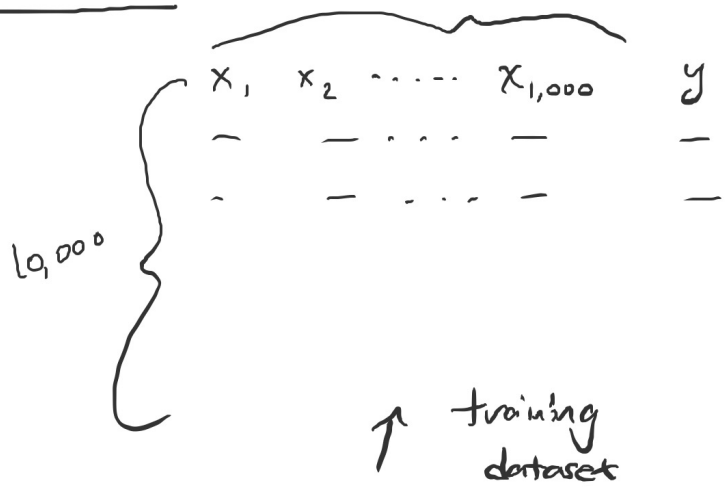
Neural Networks

Biologically inspired

Images





Neural Network

$f(x) = y$ $\quad x \rightarrow y \leftarrow$

Goal: To learn a mapping between input (x) and output (y).

	x_1	x_2	x_3	...	x_n	y
email 1						0
email 2						1
						0
email m						0

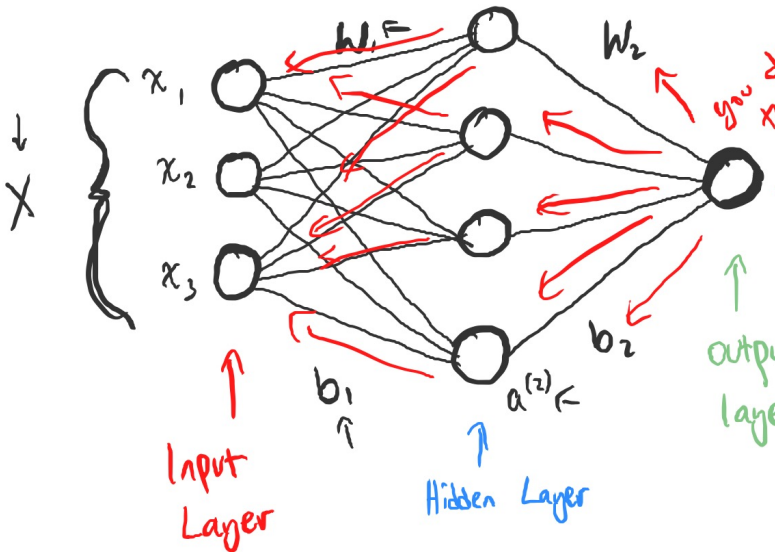
$64 \times 64 = 4096$

Logistic Regression

1. Defined model $0 \le h(x) \le 1$
2. $h(x) = g(wx + b)$
3. $J(w,b) = \frac{1}{m} \sum -(\log \hat{y})y + (\log(1-\hat{y}))$
4. Gradient Descent to update w and b.
5. $\bar{w}$ and $\bar{b}$

$h(x) = g(\bar{w}x + \bar{b})$

Forward Propagation



$\hat{y} = a^{(3)}$

J: how bad did model do?

Input Layer

Hidden Layer

output layer

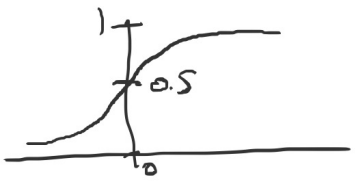
Backpropagation

Forward Propagation in Neural Network

$g(z) = \frac{1}{1+e^{-z}}$ ← Sigmoid Function

$a^{(2)} = g(W_1 x + b_1)$ ←

$\vec{W_1}, \vec{b_1}, \vec{W_2}, \vec{b_2}$



$a^{(3)} = \hat{y} = g(W_2 a^{(2)} + b_2)$

Overview of What Neural Network does.

Forward Propagation

input $x \rightarrow$ Layer 1: $a^{(1)} = W_1 x + b_1 \rightarrow$ Layer 2: $a^{(3)} = W_2 a^{(2)} + b_2 \rightarrow$

$\rightarrow \ldots \rightarrow$ Layer L: $a^{(L)} = W_L a^{(L-1)} + b_L \rightarrow \hat{y}$ ← prediction

0 - not spam

1 - spam

$J = (y - \hat{y})$

Compute Error

cycle

Go back through network and change parameters based on J.

"Fix the network"

Backpropagation

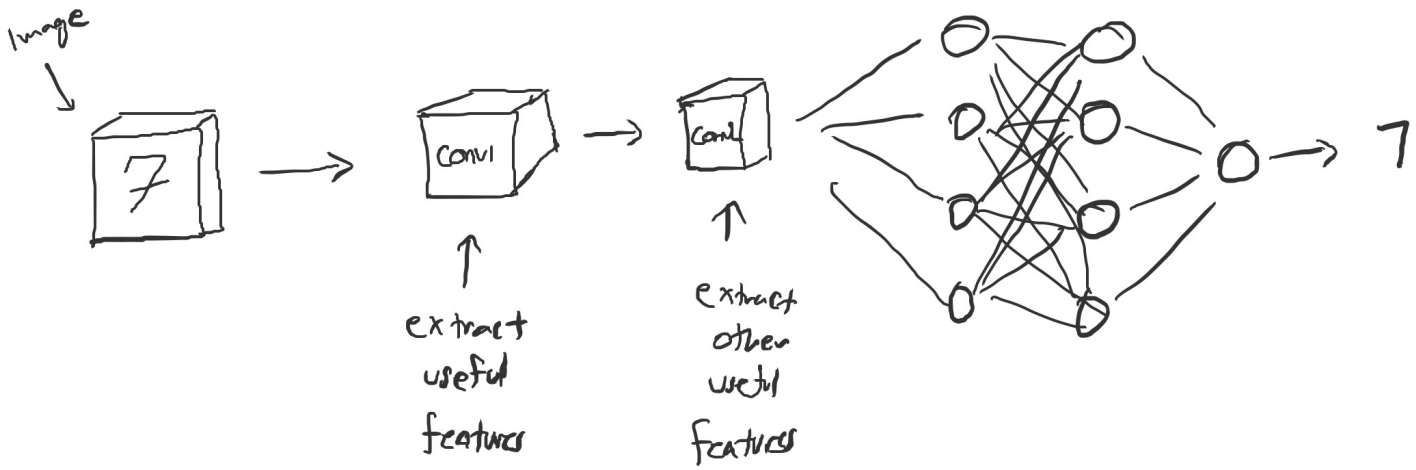


Image
7
Conv1
extract useful features
Conv2
extract other useful features
7