Utilizing Nueral Networks to Recognize Handwritten digits

founded by Warren McCullough & Walter Pitts 1944 → multi layered approach
steamed from nuero & comp sciences

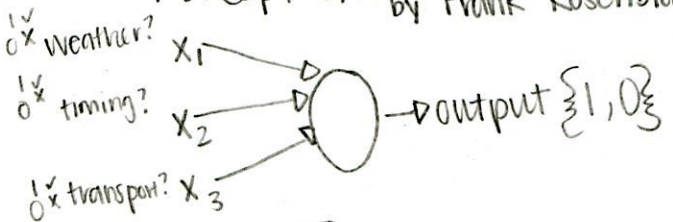
V1 = primary visual cortex, V2, V3, V4, V5
↓
140 mili nuerons

nueral networks approach → take Large # of numbers hand written & train the system
→ by increasing # of hand written samples = better accuracy

2 Types of Artifical Neurons

main

Perceptrons by Frank Rosenblatt

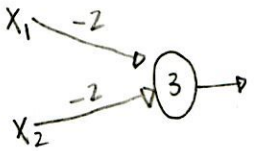


$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases}$$

Q: how to determine threshold??
$b \equiv -\text{threshold}$ (perceptron's bias)

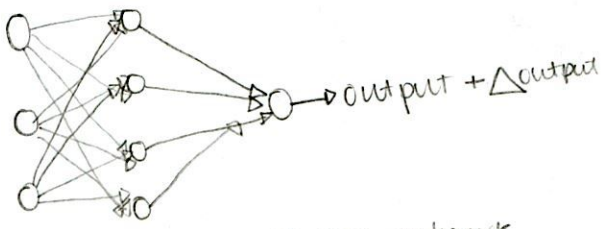
$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases}$$

ex:



① $(-2)*0 + (-2)*0 + 3 = 3 \approx 1$
② $(-2)*1 + (-2)*1 + 3 = -1 \approx 0$
③ use NAND from CSCI 250!

Sigmoid



small change to the network can cause a flip from 1 to 0

Difference: not just 1 or 0 but between 1 & 0

$\sigma(w \cdot x + b)$
↓
$\sigma(z) = \frac{1}{1+e^{-z}}$

cut up each number into 4 parts
then use the multi layered approach to train the model

big discrete & calc math problem

* utilize batch processing

reminds me of UML modeling
lots of causality cases