PERCEPTRONS, QUICKLY

NM1

2401

SL#3

KSA

NEURAL NETWORKS

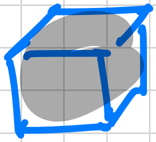
PERCEPTRON ~ BASIC BUILDING BLOCK

FRANK ROSENBLATT

- PSYCHOLOGIST
- STUDIED COGNITIVE SYSTEMS

OTHERS BEFORE HIM THOUGHT THROUGH LEARNING ALGORITHMS, BUT ... HE WROTE A COMPUTER PROGRAM THAT LEARNED TO CLASSIFY IMAGES. IN 1958

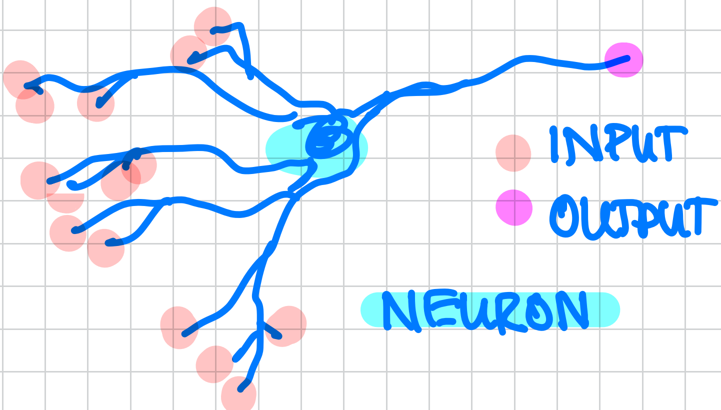
WRT CODE,

INPUT →  → OUTPUT

BLACK BOX

= MYSTERY

WRT BRAIN,



NEURON

WRT COMPUTATIONAL SELF-LEARNING,

SYSTEM OF EQNS [L05,T02]

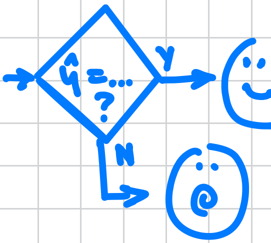
$$\left\{\begin{array}{l} x_1 \longrightarrow w_1 \mapsto x_1 w_1 = \hat{y}_1 \\ x_2 \longrightarrow w_2 \mapsto x_2 w_2 = \hat{y}_2 \\ \vdots \longrightarrow \vdots \quad \vdots \quad \vdots \\ \quad\;\; \longrightarrow \\ x_n \longrightarrow w_n \mapsto x_n w_n = \hat{y}_n \end{array}\right\} \rightarrow \sum \hat{y}_i = \hat{y}$$

PERCEPTRON

$$
\left.
\begin{array}{l}
x_1 \longrightarrow w_1 \mapsto x_1 w_1 = \hat{y}_1 \\
x_2 \longrightarrow w_2 \mapsto x_2 w_2 = \hat{y}_2 \\
\vdots \qquad \vdots \qquad \vdots \qquad \vdots \\
x_n \longrightarrow w_n \mapsto x_n w_n = \hat{y}_n
\end{array}
\right\} \to \sum \hat{y}_i = \hat{y} \quad \Rightarrow x \mapsto \hat{y}
$$

BIAS [L01, T 00|01]

$$
\left.
\begin{array}{l}
x_1 \to w_1 \mapsto x_1 w_1 \mapsto x_1 w_1 + b = z_1 \\
x_2 \to w_2 \mapsto x_2 w_2 \mapsto x_2 w_2 + b = z_2 \\
\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\
x_n \to w_n \mapsto x_n w_n \mapsto x_n w_n + b = z_n
\end{array}
\right\} \to \sum z_i = z \mapsto \hat{y} = \sigma(z) \to
$$



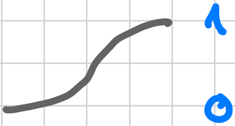
FORWARD PROPAGATION

1. $x \cdot w \Rightarrow \sum x_i \cdot w_i$
2. BIAS b
3. ACTIVATION FUNCTION

PERCEPTRONS → BINARY STEP

THIS OVERVIEW → SIGMOID

$$\sigma(z) = \frac{1}{1 - e^{-z}}$$



~S-SHAPE

LEARNING

$$\left.\begin{array}{l} x_1 \to w_1 \mapsto x_1 w_1 \mapsto x_1 w_1 + b = z_1 \\ x_2 \to w_2 \mapsto x_2 w_2 \mapsto x_2 w_2 + b = z_2 \\ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ x_n \to w_n \mapsto x_n w_n \mapsto x_n w_n + b = z_n \end{array}\right\} \to \sum z_i = z \mapsto \hat{y} = \sigma(z)$$

$r = \hat{y} - y$

ADJUST w_i, b

1. BACKPROPAGATION ~ BACKWARD PROPAGATION.

[L2, T00|02]

→ BACKWARD ERROR : Δ OUTPUT → EXPECTED Δ INPUT

→ ERROR: $SE = (\hat{y}_i - y_i)^2 \Rightarrow MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 = C$

→ C ~ COST FUNCTION ~ LOSS FUNCTION

2. OPTIMIZATION ~ FIND BEST w_i, b.

→ USE GRADIENT DESCENT. } [L06, T02|03]

CONJUGATE GRADIENT

$$\frac{\partial C}{\partial w_i} = \frac{\partial C}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z} \cdot \frac{\partial z}{\partial w_i}$$

$$\Rightarrow \frac{\partial C}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}} \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = 2 \cdot \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i) = \frac{2}{n} \operatorname{sum}(\hat{y}_i - y_i)$$

$$\frac{\partial \hat{y}}{\partial z} = \frac{\partial}{\partial z}\sigma(z) = \frac{\partial}{\partial z}\left(\frac{1}{1-e^{-z}}\right) = \frac{e^{-z}}{(1+e^{-z})^2}$$

$$= \frac{1}{1+e^{-z}} \cdot \frac{e^{-z}}{1+e^{-z}} = \frac{1}{1+e^{-z}} \cdot \left(1 - \frac{1}{1+e^{-z}}\right) = \sigma(z)(1-\sigma(z))$$

$$\frac{\partial z}{\partial w_j} = \frac{\partial}{\partial w_j}(z) = \frac{\partial}{\partial w_i} \sum_{i=1}^{n} (x_i w_i + b) = x_i$$

$$\Rightarrow \frac{\partial C}{\partial w_i} = \frac{2}{n} \cdot \text{sum}(\hat{y}_i - y_i) \cdot \sigma(z)(1-\sigma(z)) \cdot x_i$$

$$\frac{\partial C}{\partial b} = \frac{2}{n} \cdot \text{sum}(\hat{y}_i - y_i) \cdot \sigma(z)(1-\sigma(z))$$

SORRY! I SHOULD'VE EXPLICITLY SAID THIS, TOO:

$$w_{i,\text{NEXT}} = w_{i,\text{PREV}} - \alpha \cdot \frac{\partial C}{\partial w_i}$$

$$b_{\text{NEXT}} = b_{\text{PREV}} - \alpha \cdot \frac{\partial C}{\partial b}$$

WHERE HYPERPARAMETER α IS THE LEARNING RATE.