2024/07/05

Exercise 3

Consider a perceptron that accepts complex inputs x_1 and x_2. The weights w_1 and w_2 are also complex numbers, and the threshold is zero. The perceptron fires if the condition $\Re(x_1w_1 + x_2w_2) \geq \Im(x_1w_1 + x_2w_2)$ is satisfied. The binary input 0 is coded as the complex number $(1,0)$ and the binary input 1 as the number $(0,1)$. How many of the logical functions of two binary arguments can be computed with this system? Can XOR be computed?

$x_1, x_2, w_1, w_2 \in \mathbb{C}$ $\qquad R(x_1w_1 + x_2w_2) \geq Im(x_1w_1 + x_2w_2)$

· Binary inputs $\quad x_1, x_2 \in \{(1,0), (0,1)\}$

$\emptyset \Rightarrow 1+0i$

$1 \Rightarrow 0+i$

↳ $(0,0)$ $(0,1)$ $(1,0)$ $(1,1)$

· Activation function

$z = x_1w_1 + x_2w_2 \qquad R(z) \geq Im(z)$ → linear decision boundary in complex plane

· How many boolean functions? $\quad 2^{2^2} = 16$ Boolean functions for binary inputs

however a linear classifier can't separate all of them

→ For each input pair compute $= x_1w_1 + x_2w_2$

Apply firing condition

Vary $w_1, w_2 \in \mathbb{C}$ to realize different labelings on the 4 inputs

↳ Binary classifier over 4 fixed complex vectors where decision regions are based on

$R(z) - Im(z) \geq 0$ $\quad$ i.e. sign of $R(z) - Im(z)$

So we're projecting into the real plane using the map

$x_1w_1 + x_2w_2 \to R - Im$ $\quad$: Single linear inequality over 4 complex valued points

(similar to a real valued perceptron over 2D inputs)

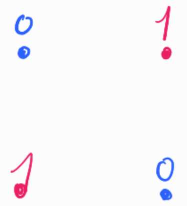
~ From perceptron theory

· Number of labelings (dichotomies) that a linear classifier can realize over 4 inputs is at most 14

- XOR is not linearly separable

- Check the XOR function

		XOR output
$(0,0) \to$	$(1,1)$	$\rightsquigarrow 0$
$(0,1) \to$	$(1,i)$	$\rightsquigarrow 1$
$(1,0) \to$	$(i,1)$	$\rightsquigarrow 1$
$(1,1) \to$	(i,i)	$\rightsquigarrow 0$



↳ Can't be linearly separable.

XOR can't be computed
- A perceptron with any linear rule can only split inputs with line or plane
- Complex perceptron still use a linear function on inputs to decide.