



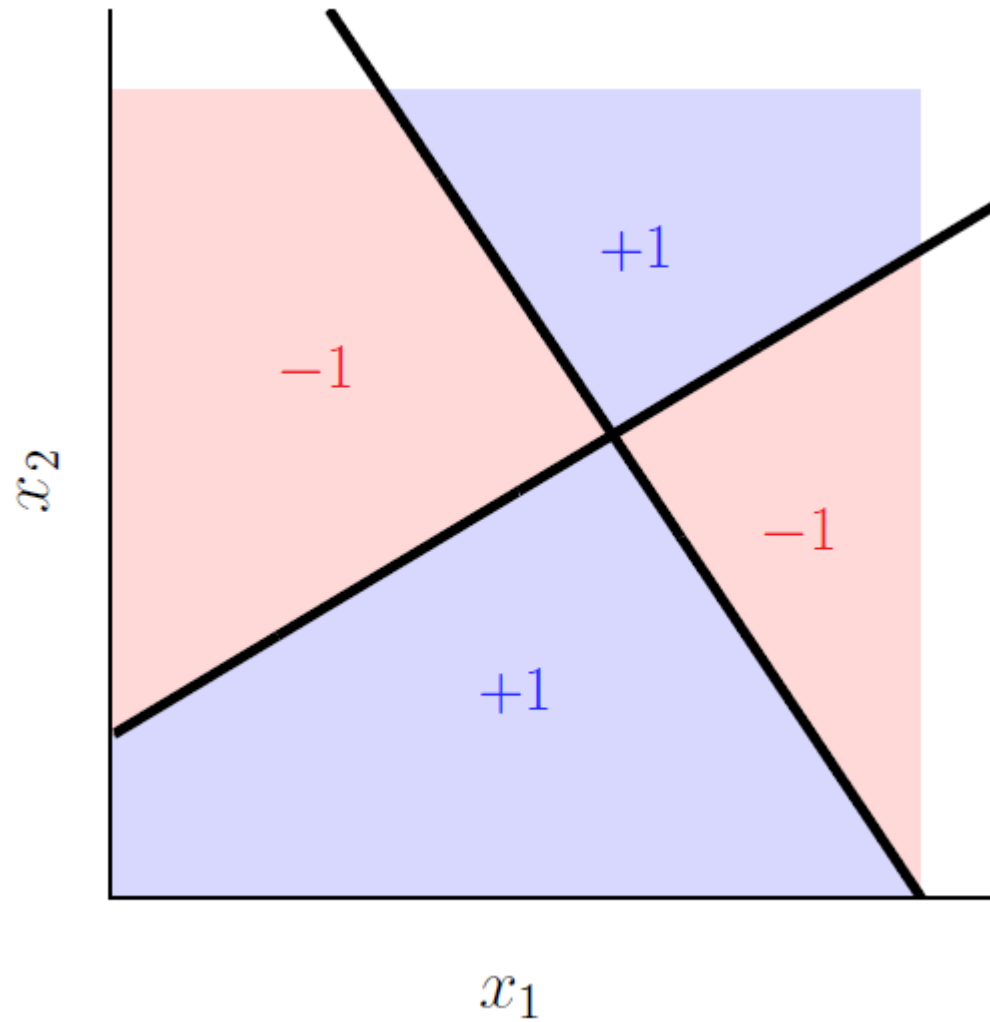
IIC 2433 Minería de Datos

<https://github.com/marcelomendoza/IIC2433>

- MLP -

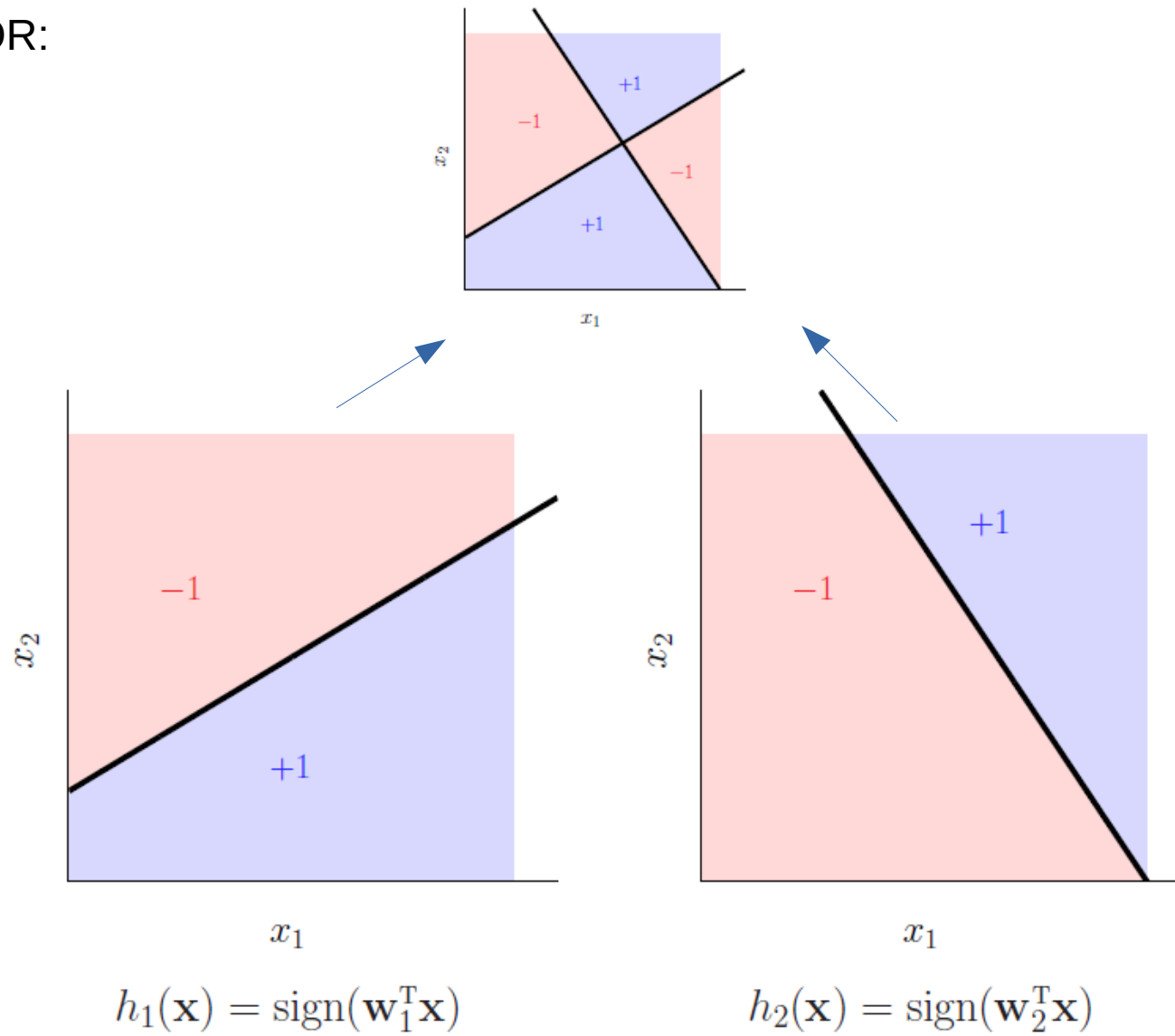
La limitación del modelo lineal

XOR:



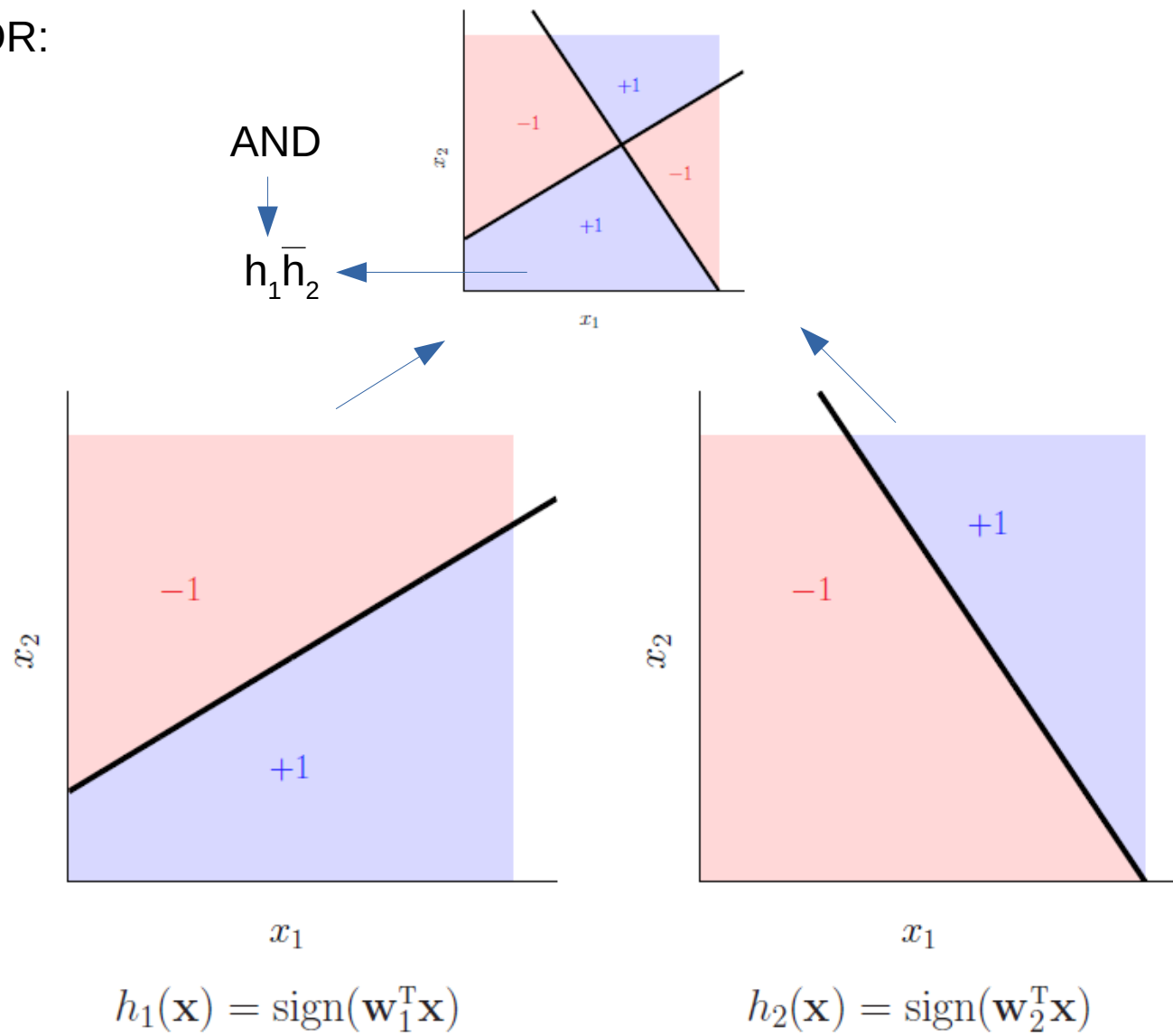
Combinación de perceptrones

XOR:



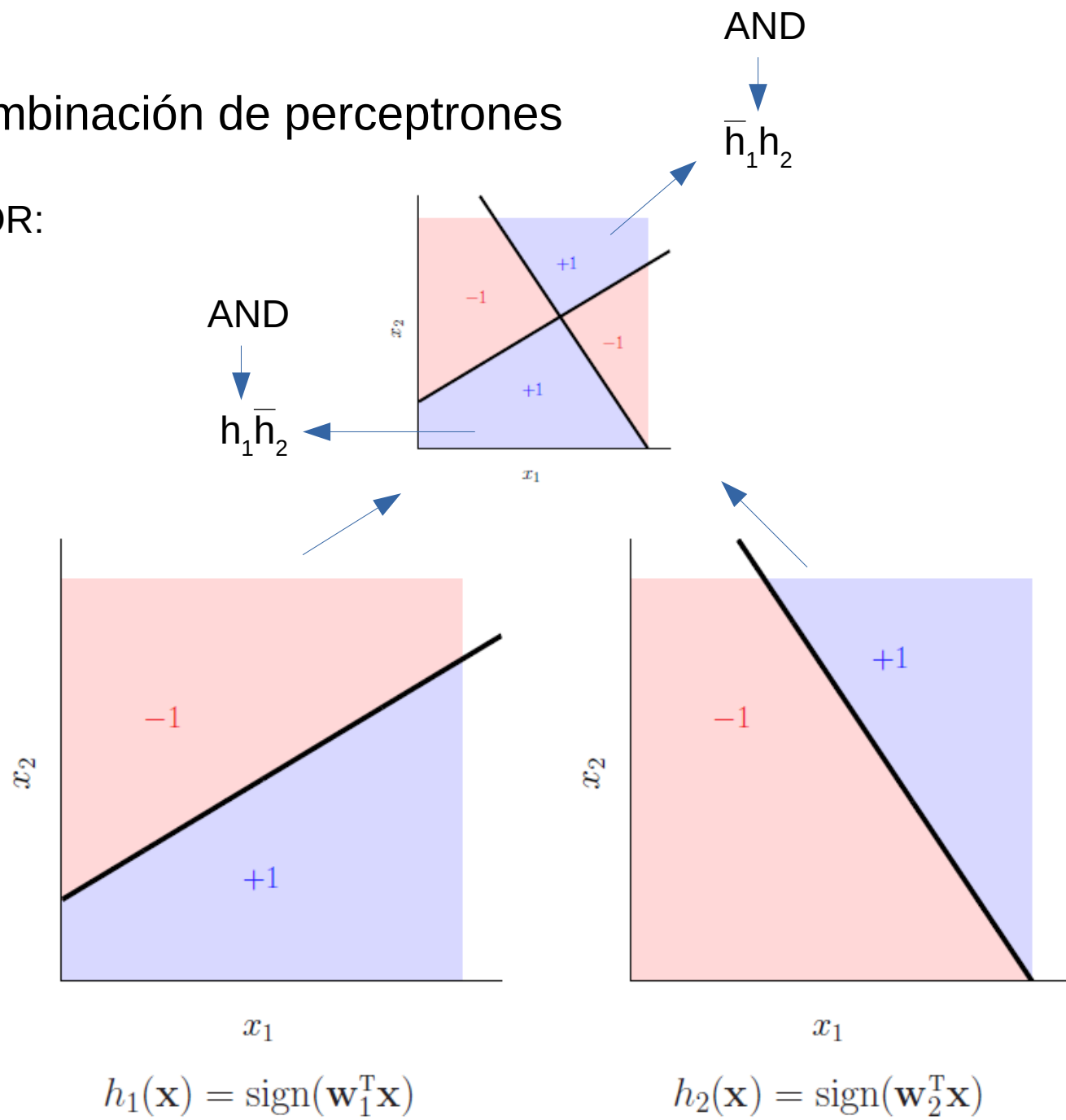
Combinación de perceptrones

XOR:



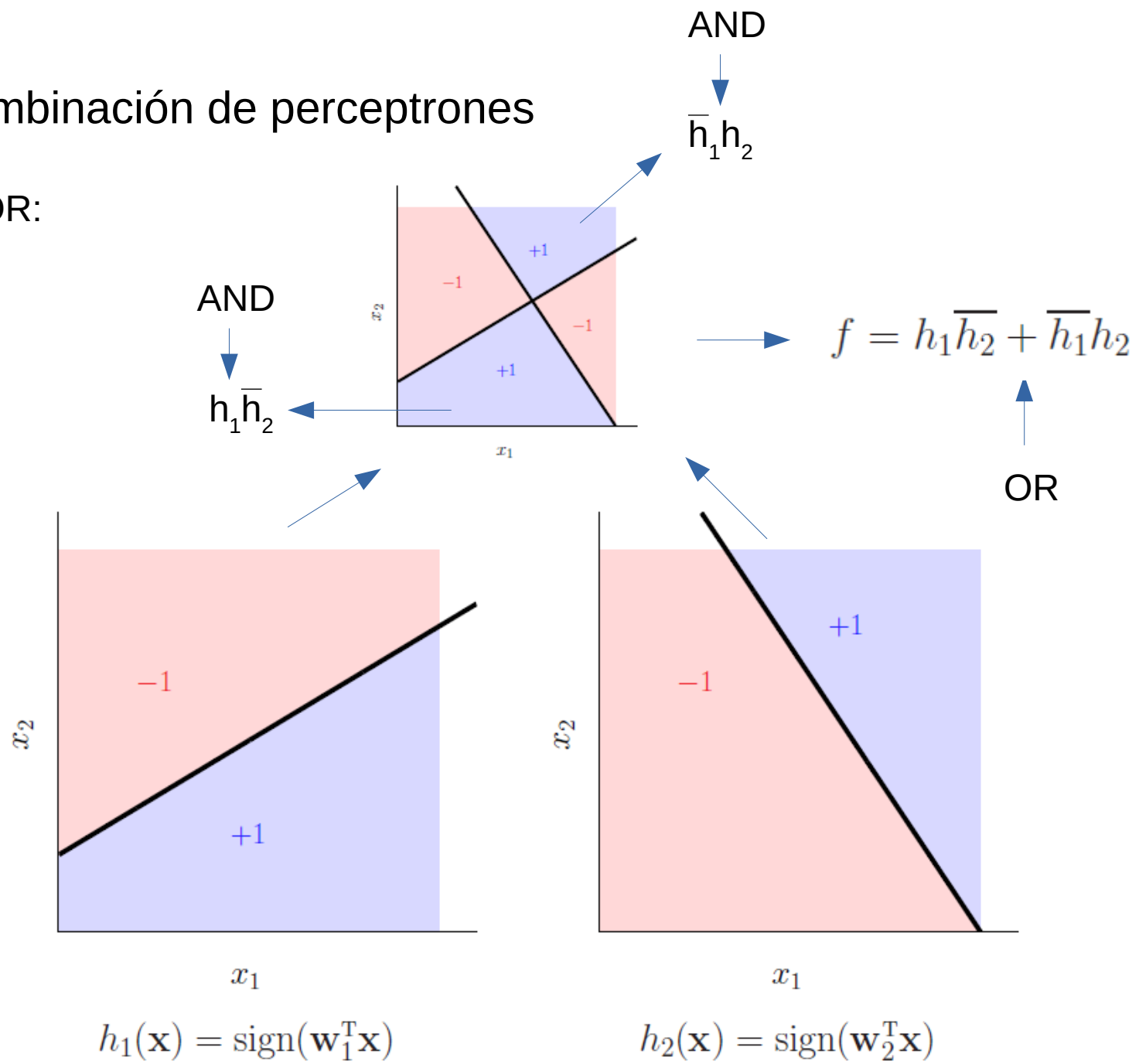
Combinación de perceptrones

XOR:



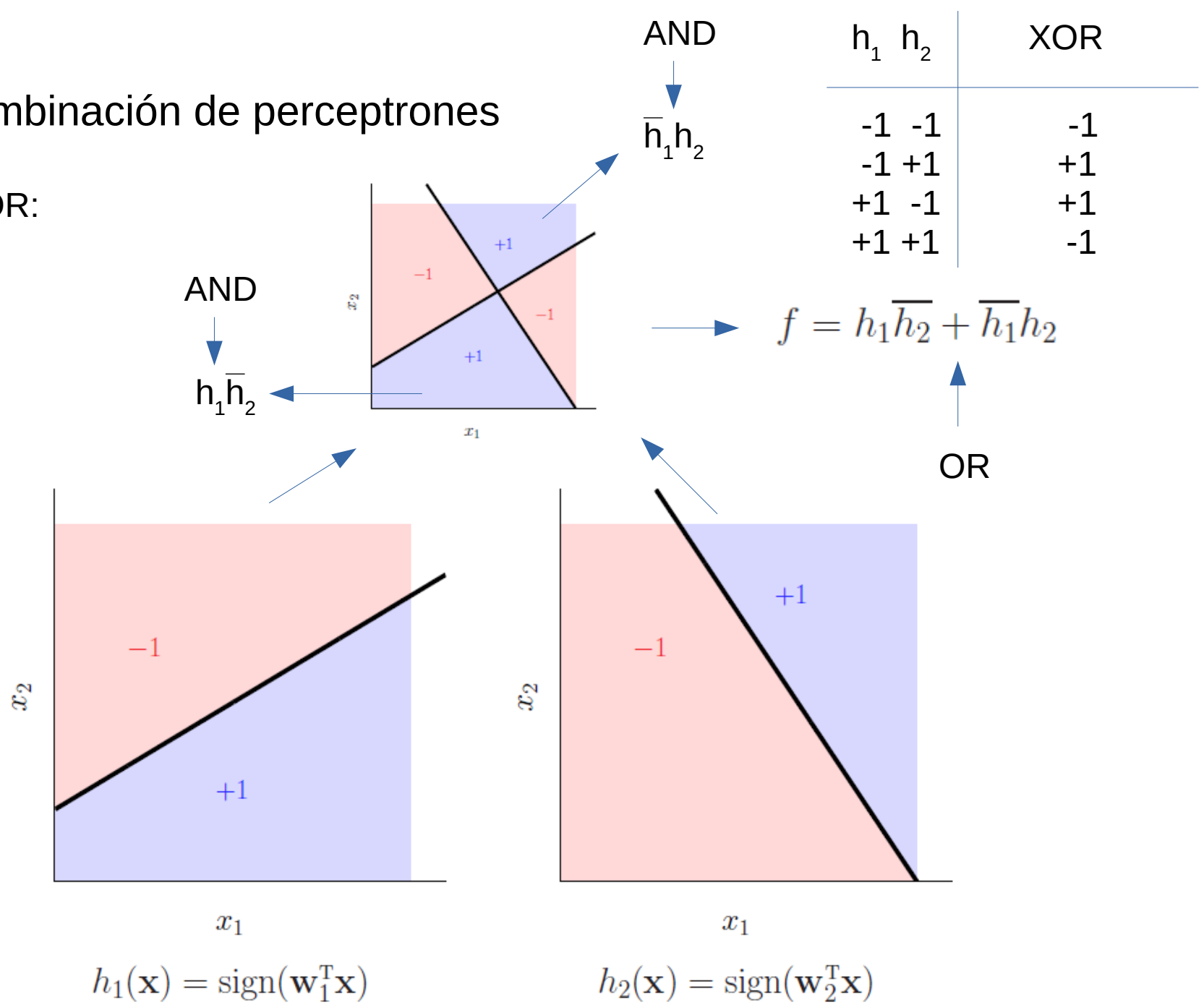
Combinación de perceptrones

XOR:



Combinación de perceptrones

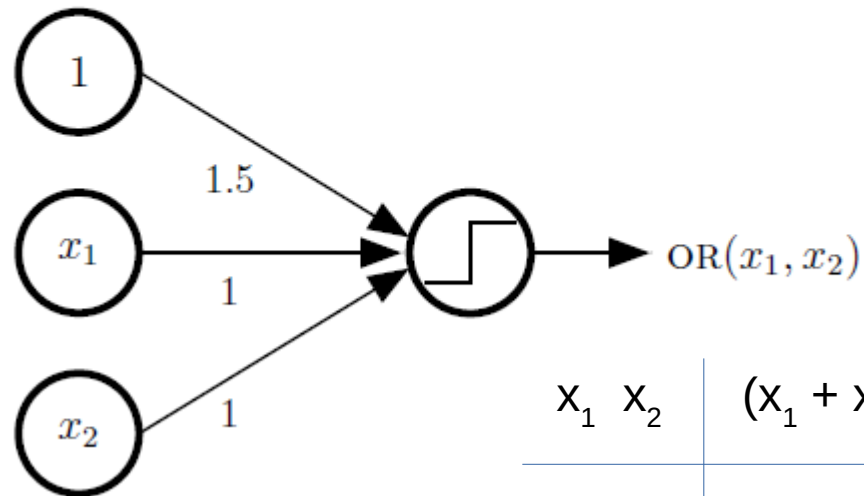
XOR:



Combinación de perceptrones

OR:

$$\text{OR}(x_1, x_2) = \text{sign}(x_1 + x_2 + 1.5)$$

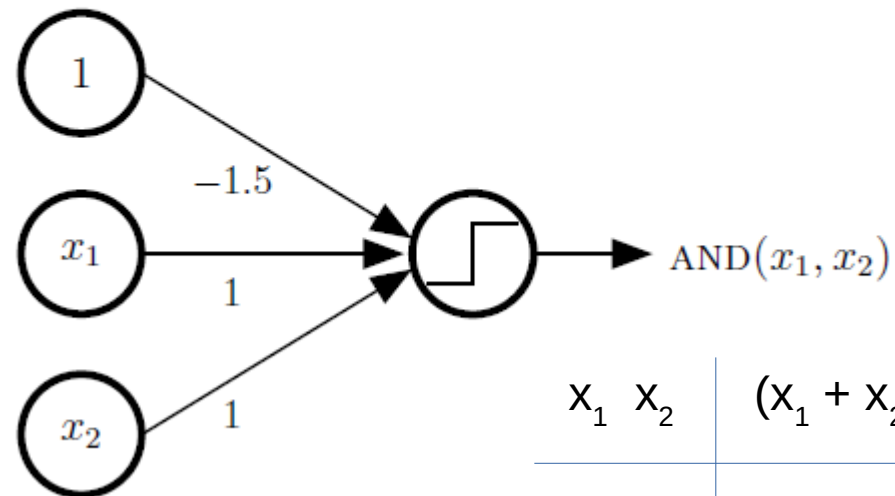


x_1	x_2	$(x_1 + x_2 + 1.5)$	signo
-1	-1	-0.5	-1
-1	+1	1.5	+1
+1	-1	1.5	+1
+1	+1	3.5	+1

Combinación de perceptrones

AND:

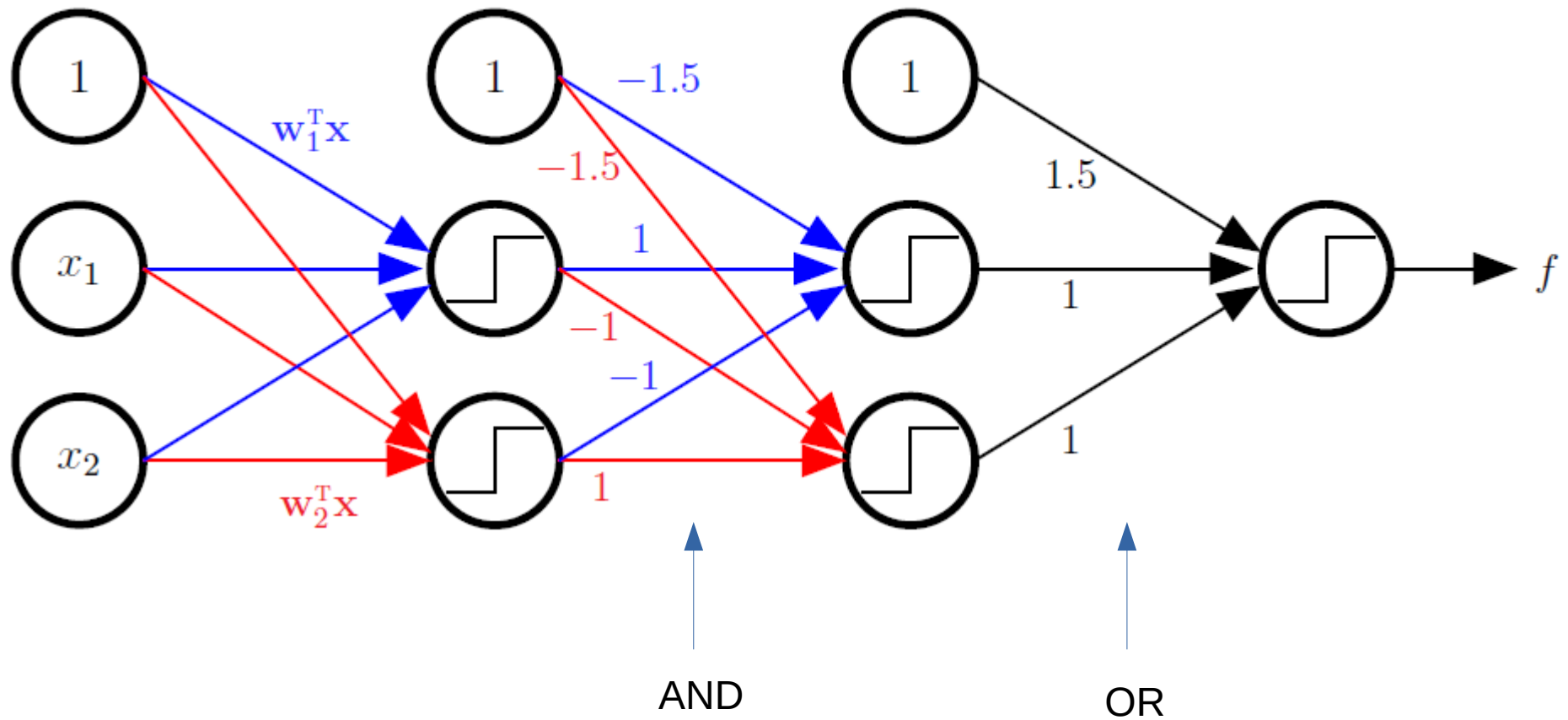
$$\text{AND}(x_1, x_2) = \text{sign}(x_1 + x_2 - 1.5)$$



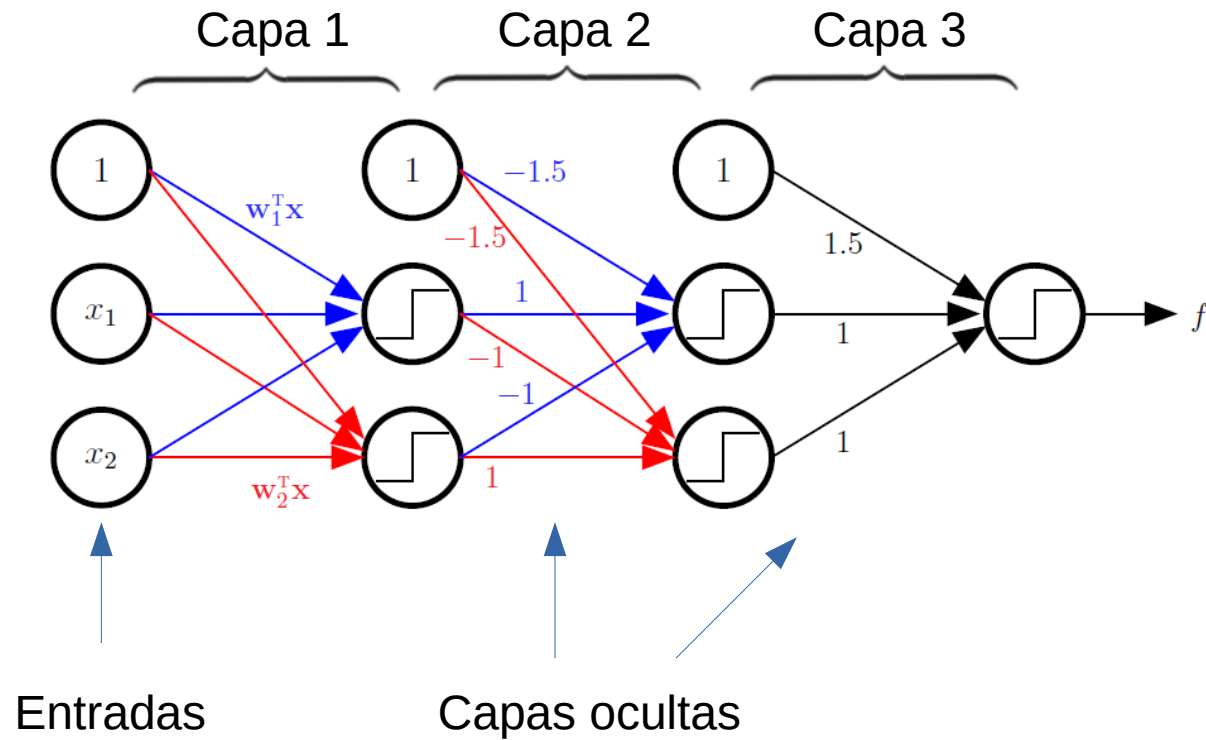
x_1	x_2	$(x_1 + x_2 - 1.5)$	signo
-1	-1	-3.5	-1
-1	+1	-1.5	-1
+1	-1	-1.5	-1
+1	+1	0.5	+1

Combinación de perceptrones

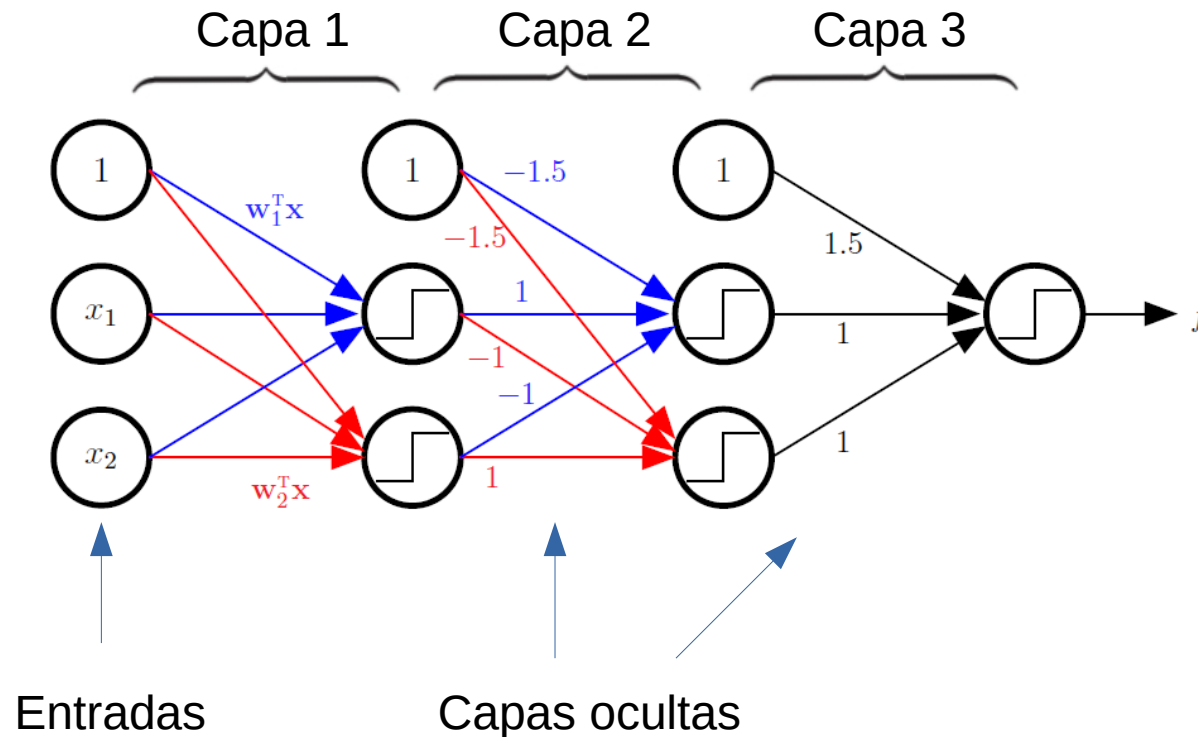
$$f = h_1 \overline{h_2} + \overline{h_1} h_2$$



El perceptrón multicapa (Multi-Layer Perceptron)



El perceptrón multicapa (Multi-Layer Perceptron)

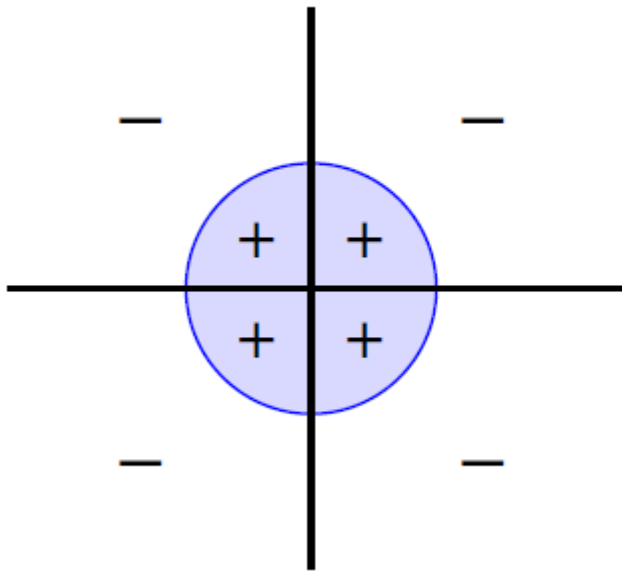


Aproximación universal

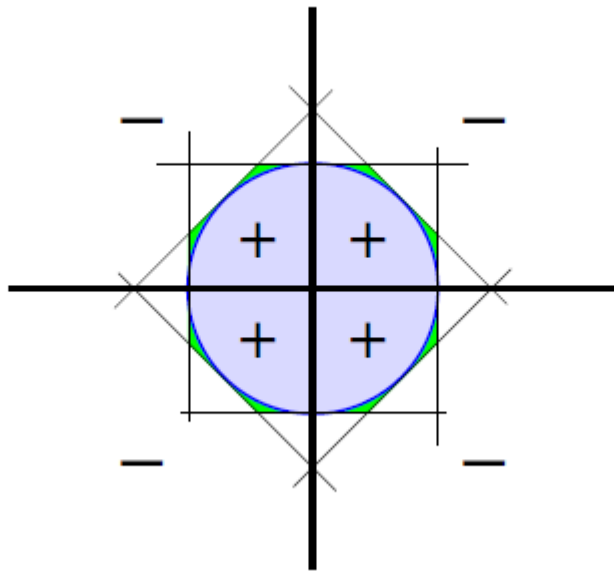
Cualquier función que se puede descomponer en separadores lineales puede ser implementada por un MLP de 3 capas

El perceptrón multicapa (Multi-Layer Perceptron)

Un separador suave puede ser aproximado por N separadores lineales.



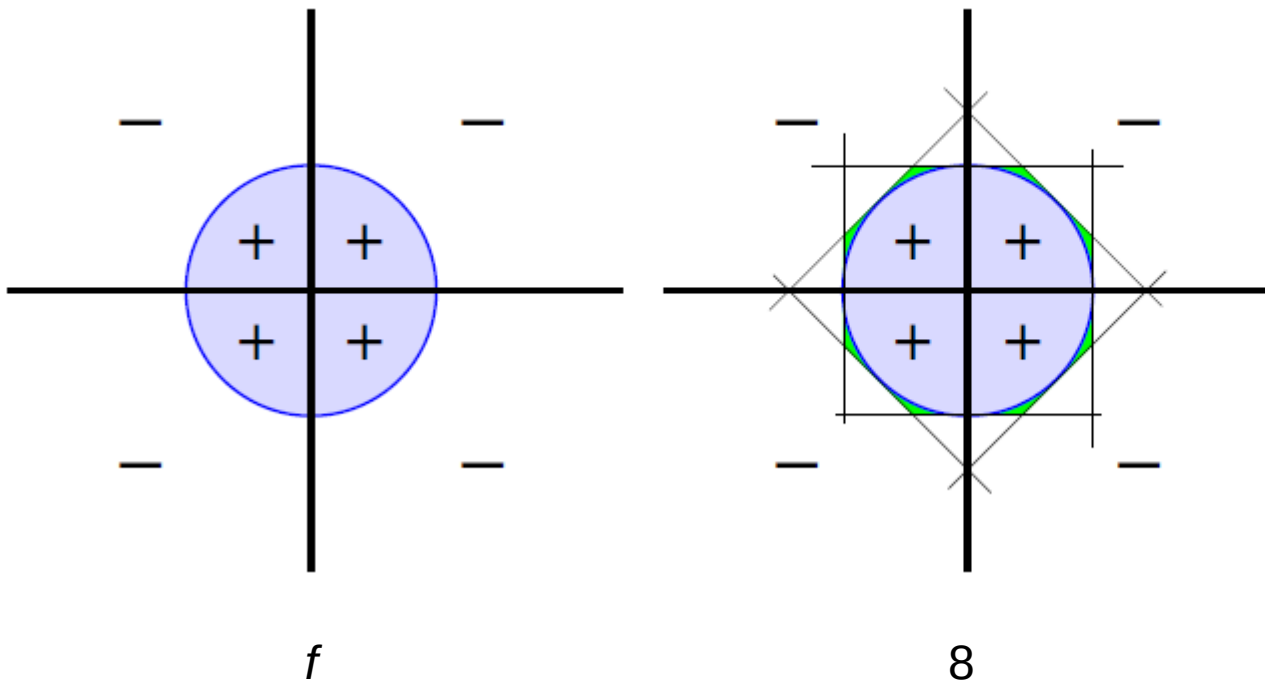
f



¿Cuántos perceptrones hay?

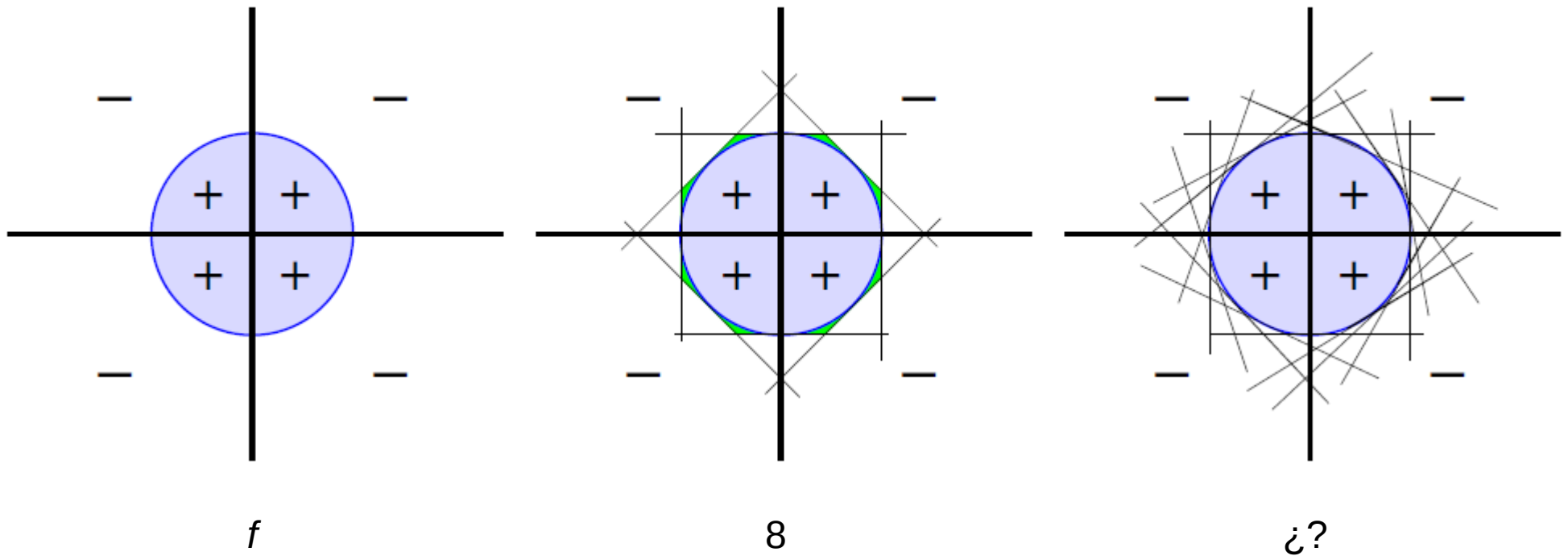
El perceptrón multicapa (Multi-Layer Perceptron)

Un separador suave puede ser aproximado por N separadores lineales.



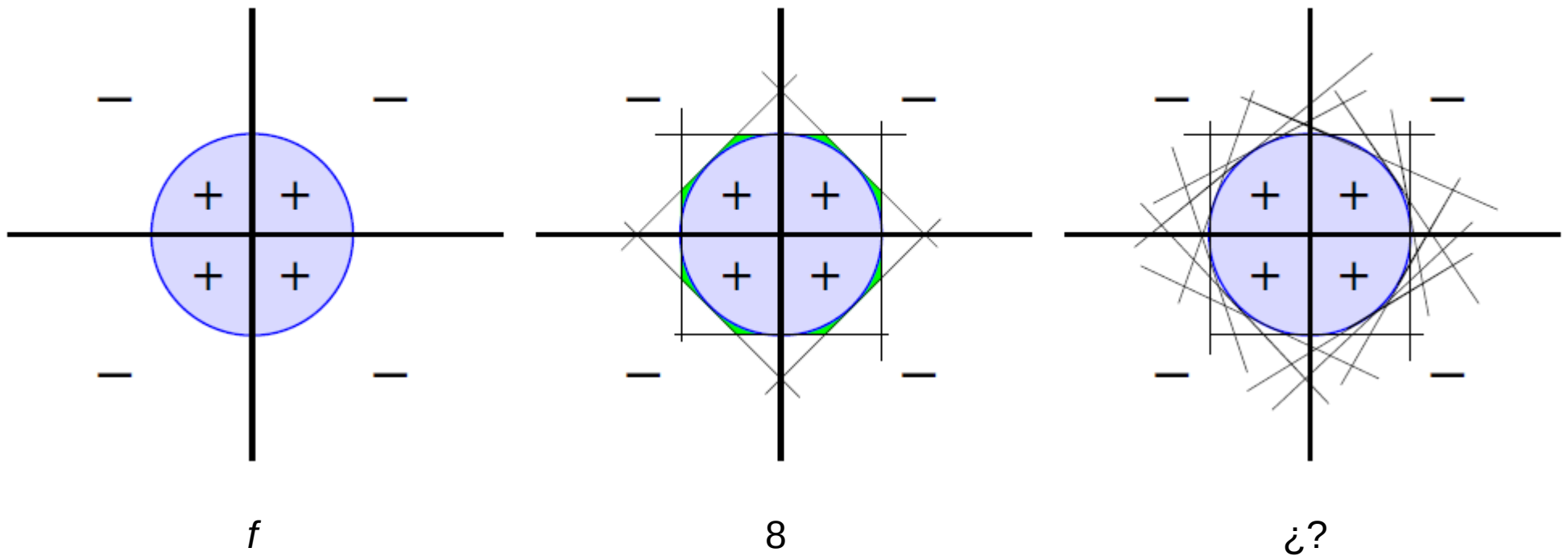
El perceptrón multicapa (Multi-Layer Perceptron)

Un separador suave puede ser aproximado por N separadores lineales.



El perceptrón multicapa (Multi-Layer Perceptron)

Un separador suave puede ser aproximado por N separadores lineales.



Tradeoff aproximación-generalización:

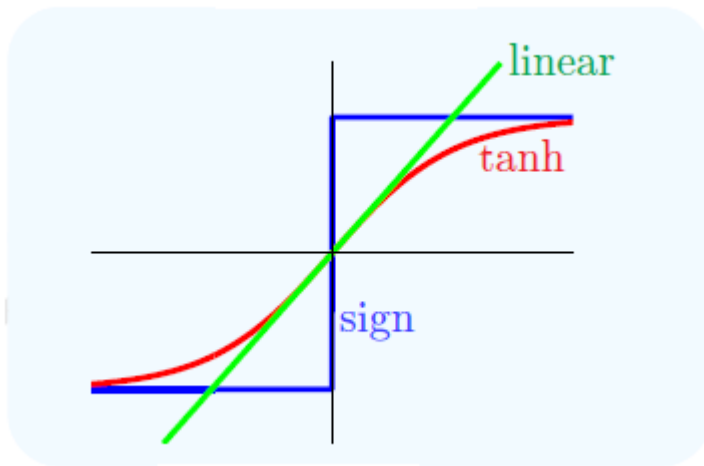
Más neuronas, mejor aproximación.

Más neuronas, peor generalización.

El perceptrón multicapa (Multi-Layer Perceptron)

Para entrenar un MLP necesitamos reemplazar la función signo dado que no es diferenciable.

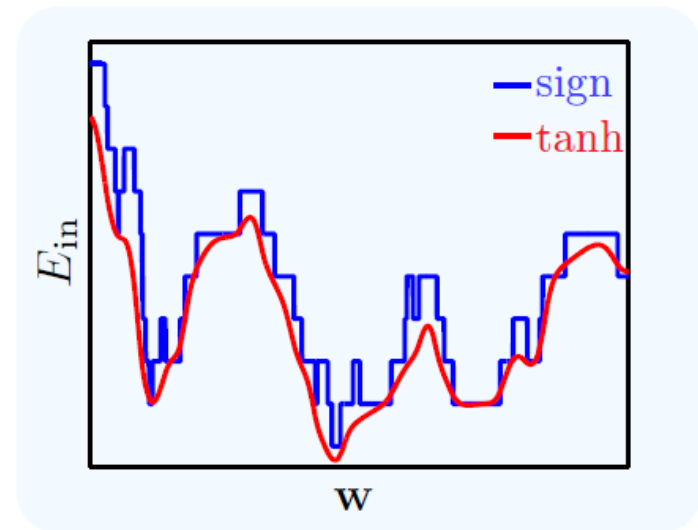
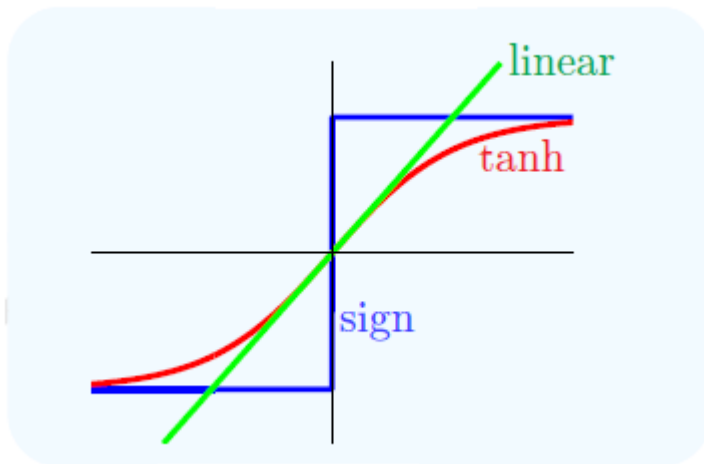
Podemos suavizar la función signo con la tangente hiperbólica, que tiene un comportamiento lineal cerca del origen y es cercana a +1 o -1 para entradas grandes.



El perceptrón multicapa (Multi-Layer Perceptron)

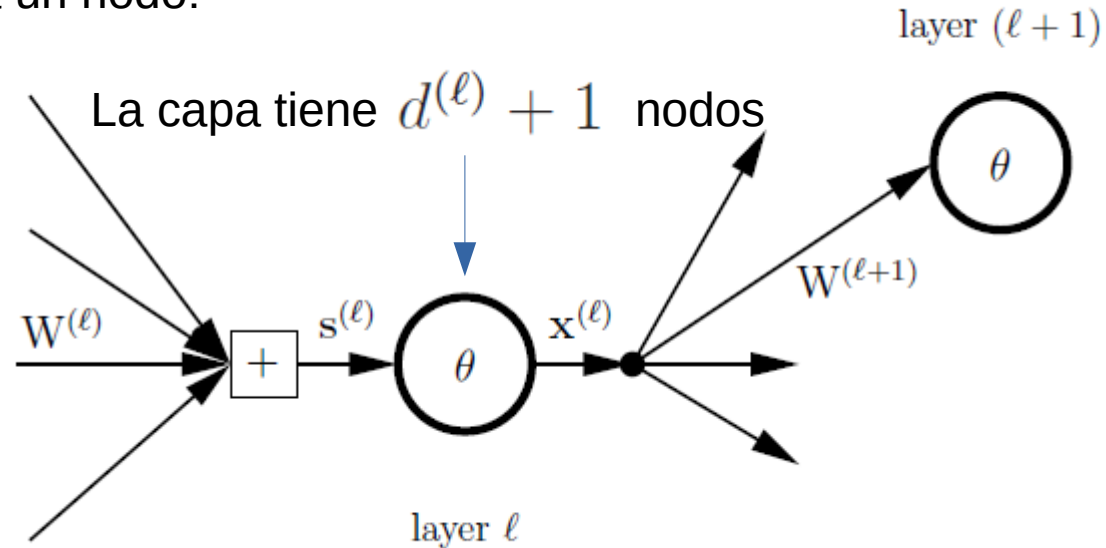
Para entrenar un MLP necesitamos reemplazar la función signo dado que no es diferenciable.

Podemos suavizar la función signo con la tangente hiperbólica, que tiene un comportamiento lineal cerca del origen y es cercana a +1 o -1 para entradas grandes.



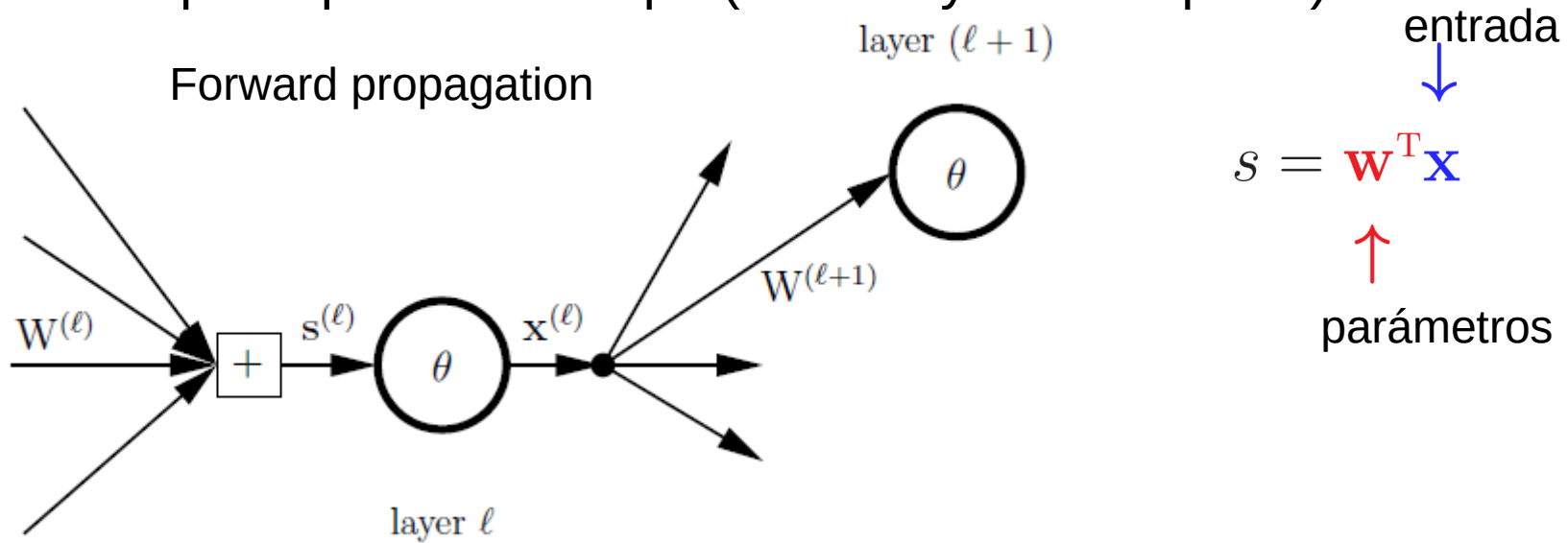
El perceptrón multicapa (Multi-Layer Perceptron)

Zoom in a un nodo:

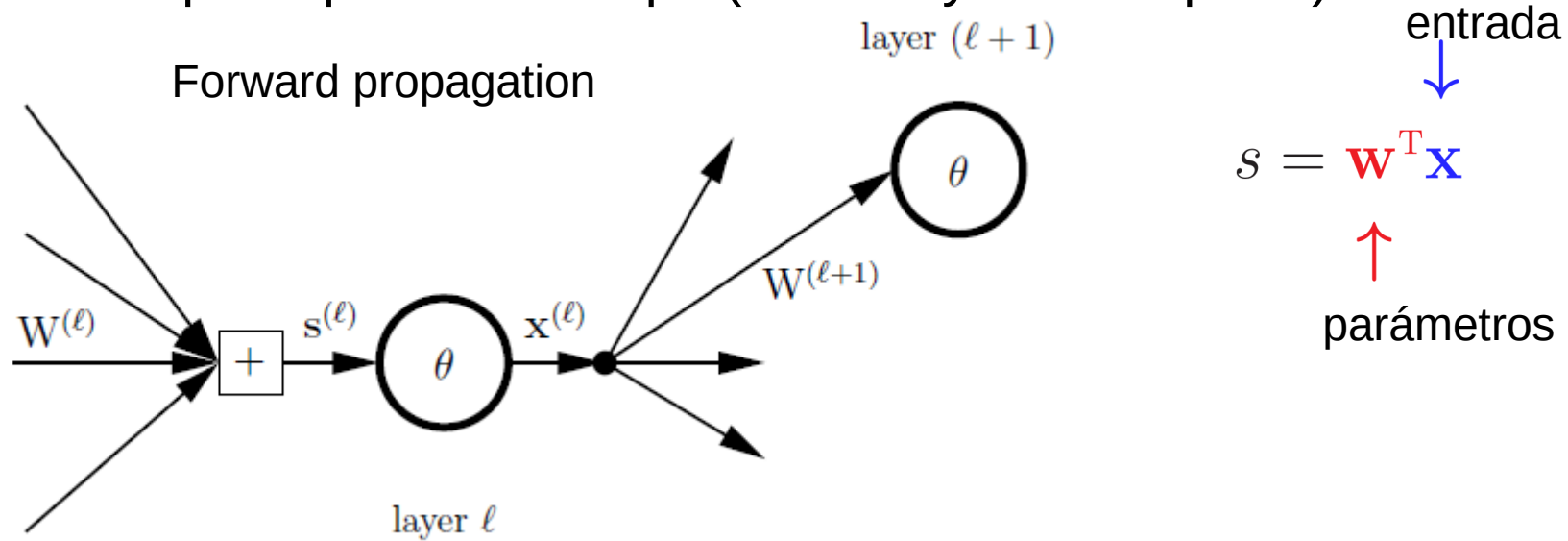


entradas	$\mathbf{s}^{(\ell)}$	$d^{(\ell)}$ dimensional
salidas	$\mathbf{x}^{(\ell)}$	$d^{(\ell)} + 1$ dimensional
pesos de entrada	$W^{(\ell)}$	$(d^{(\ell-1)} + 1) \times d^{(\ell)}$ dimensional
pesos de salida	$W^{(\ell+1)}$	$(d^{(\ell)} + 1) \times d^{(\ell+1)}$ dimensional

El perceptrón multicapa (Multi-Layer Perceptron)



El perceptrón multicapa (Multi-Layer Perceptron)

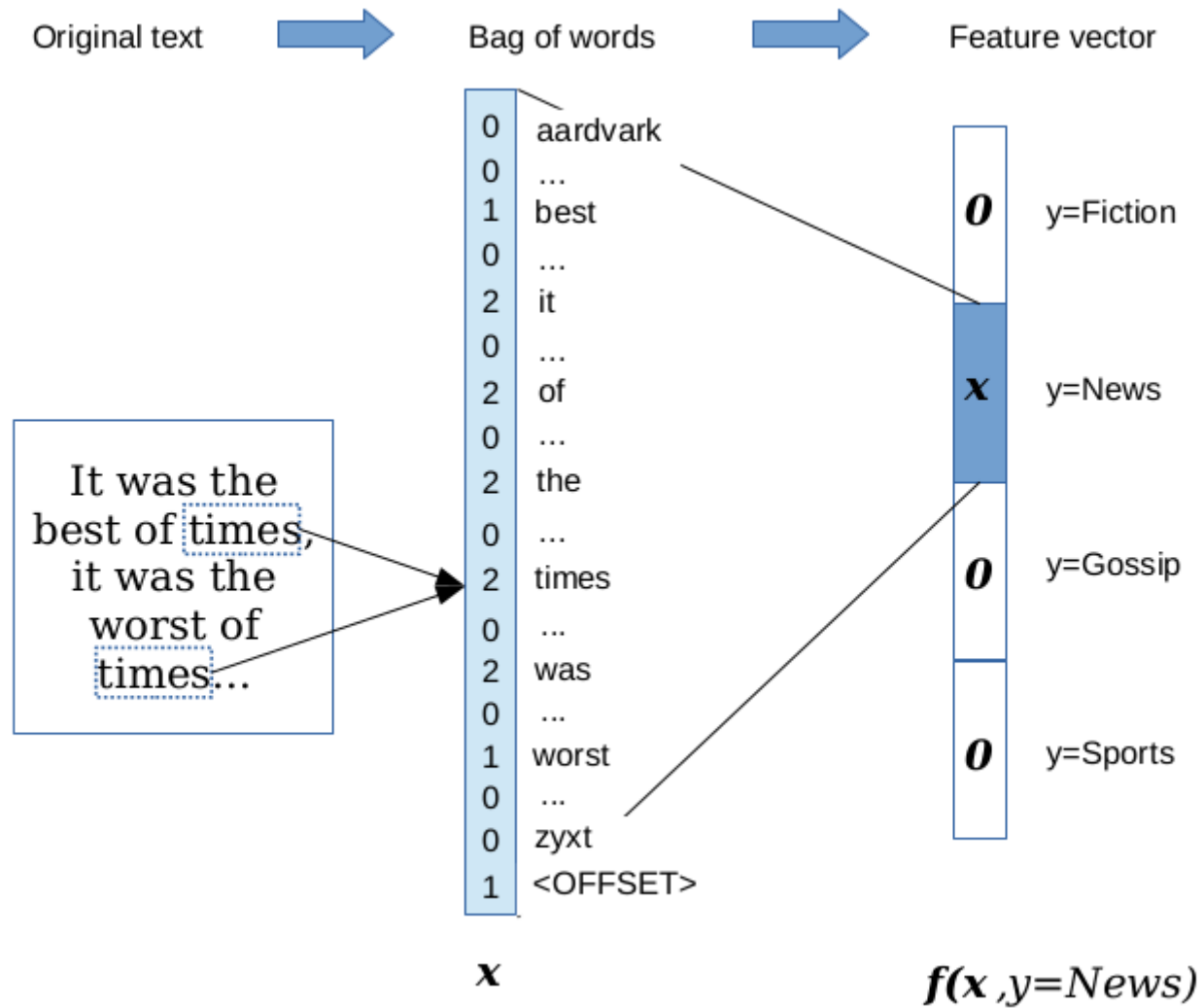


$$\mathbf{s}^{(\ell)} = (\mathbf{W}^{(\ell)})^T \mathbf{x}^{(\ell-1)}$$

$$\begin{bmatrix} s_1^{(\ell)} \\ s_2^{(\ell)} \\ \vdots \\ s_j^{(\ell)} \\ \vdots \\ s_{d^{(\ell)}}^{(\ell)} \end{bmatrix} = \begin{bmatrix} (\mathbf{w}_1^{(\ell)})^T \\ (\mathbf{w}_2^{(\ell)})^T \\ \vdots \\ (\mathbf{w}_j^{(\ell)})^T \\ \vdots \\ (\mathbf{w}_{d^{(\ell)}}^{(\ell)})^T \end{bmatrix} \mathbf{x}^{(\ell-1)}$$

- CLASIFICACIÓN DE TEXTO -

BOW



Clasificación con word embeddings

```
inputs = Input(shape=(max_tokens, ))

embeddings_layer = Embedding(input_dim=len(tokenizer.index_word)+1,
                              output_dim=embed_len, input_length=max_tokens, trainable=False,
                              weights=[glove_50_embeddings])

dense1 = Dense(128, activation="relu")
dense2 = Dense(64, activation="relu")
dense3 = Dense(len(classes), activation="softmax")

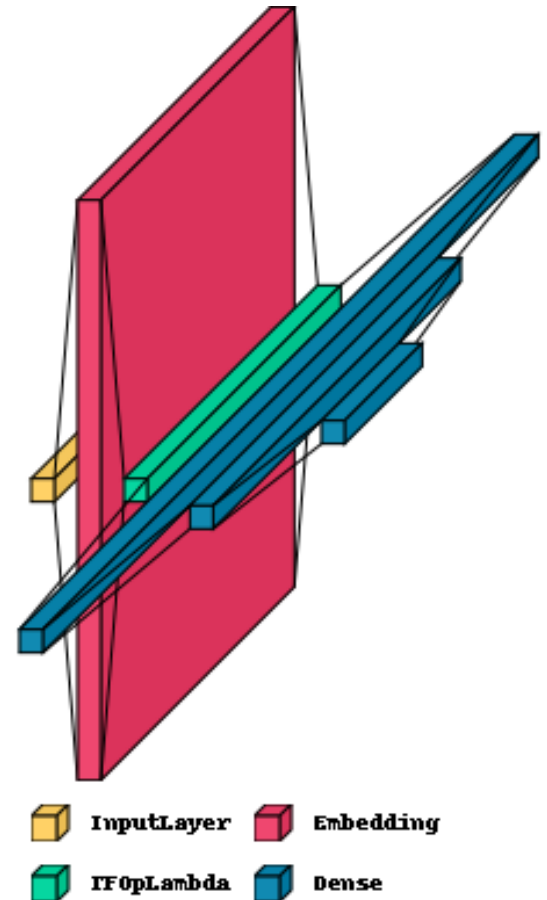
x = embeddings_layer(inputs)
x = tensorflow.reduce_sum(x, axis=1)
x = dense1(x)
x = dense2(x)
outputs = dense3(x)

model = Model(inputs=inputs, outputs=outputs)
```

| V |



forward



Clasificación con word embeddings

