

# Detección de patrones de difracción, utilizando el paradigma de redes neuronales

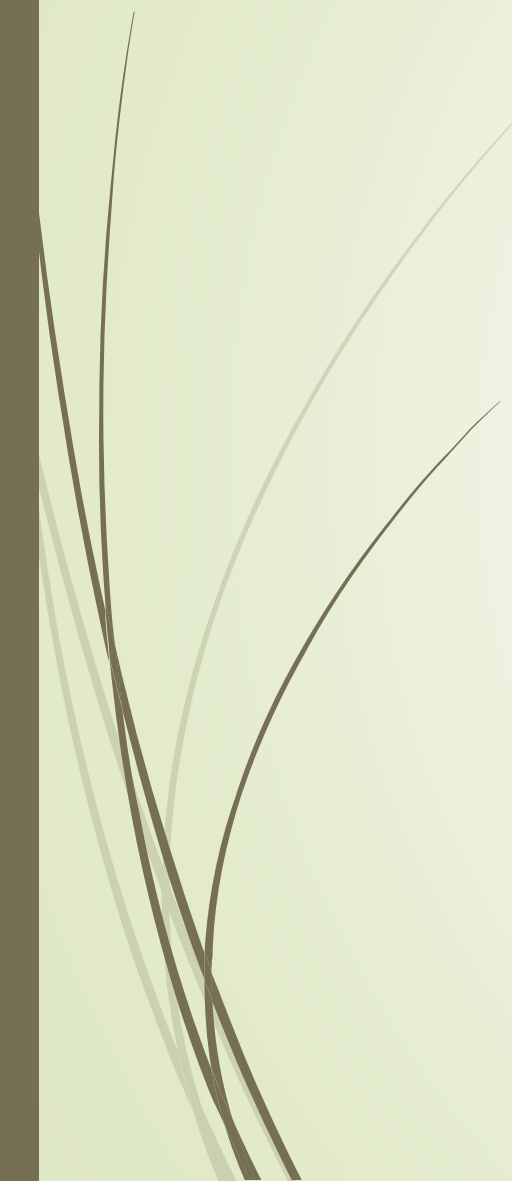
Estudiante: Axel Tapia

Asesor: Dr. Benjamín Hernández





# Contenido

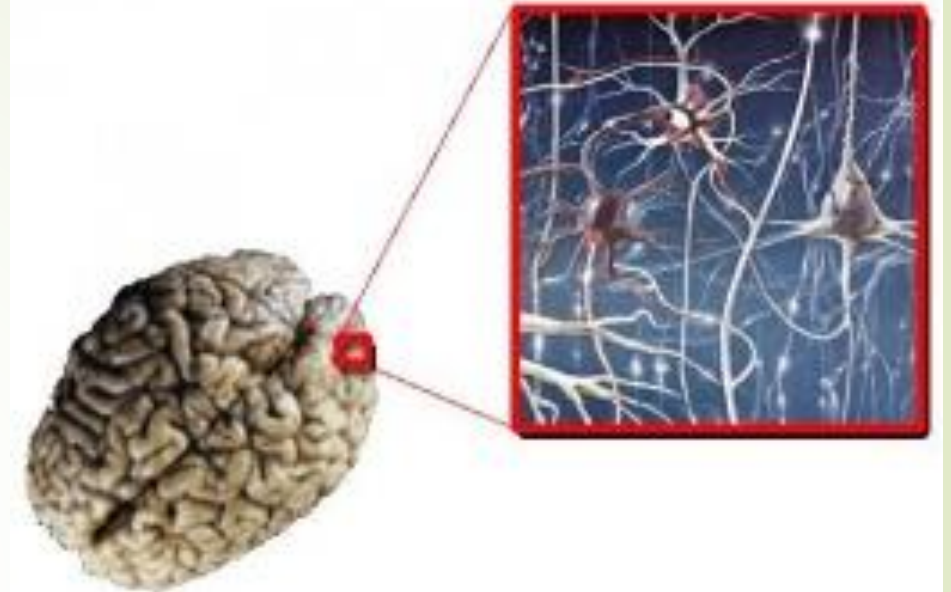
- Inspiración biológica
  - Conceptos básicos
  - Ejemplo
  - Uso en TAOS2
  - Experimentación y resultados
  - Conclusiones
  - Referencias
- 

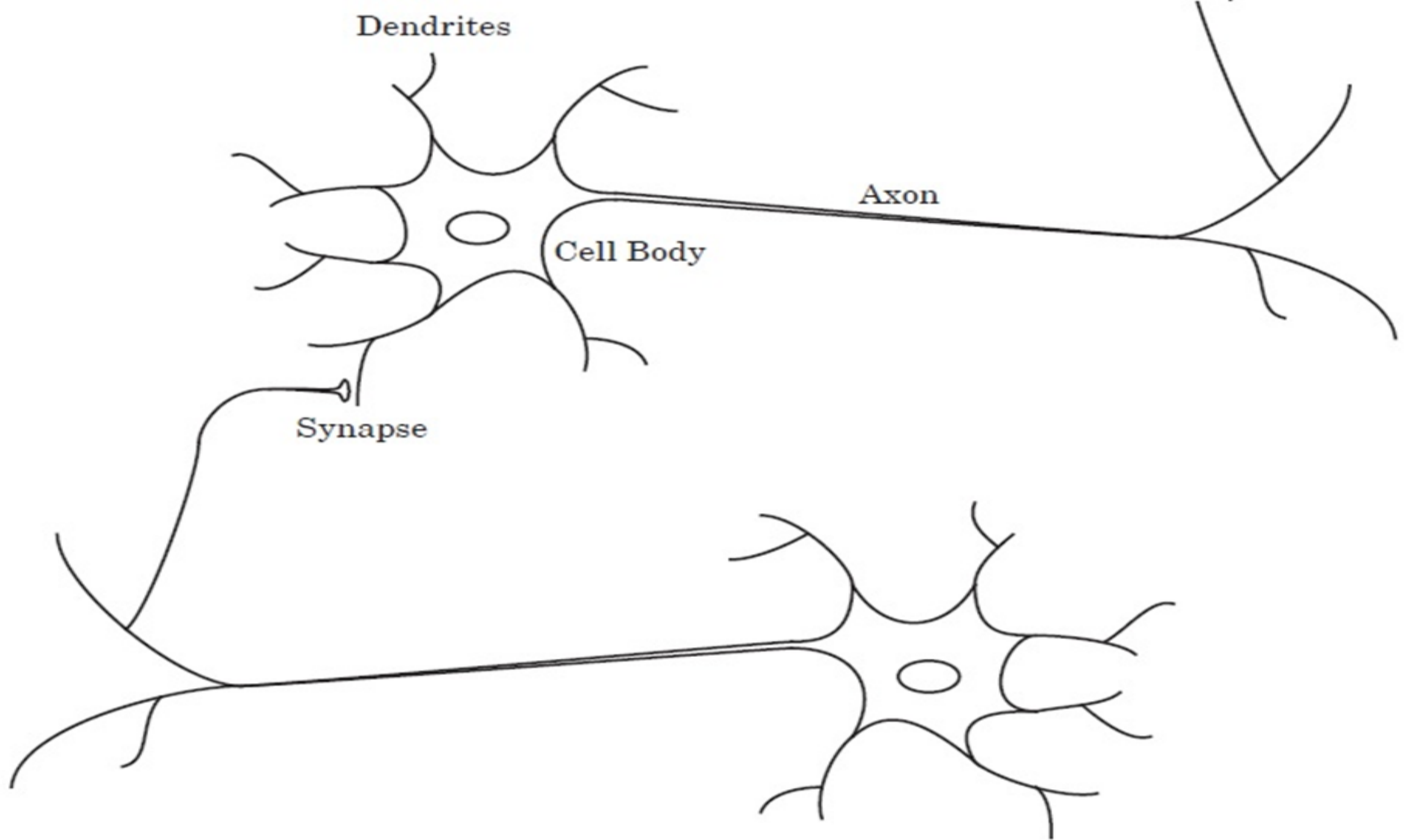


# Inspiración biológica

## Cerebro humano

- $\sim 10^{11}$  neuronas
- $\sim 10^4$  conexiones por neurona
- $\sim 10^{15}$  conexiones en la red



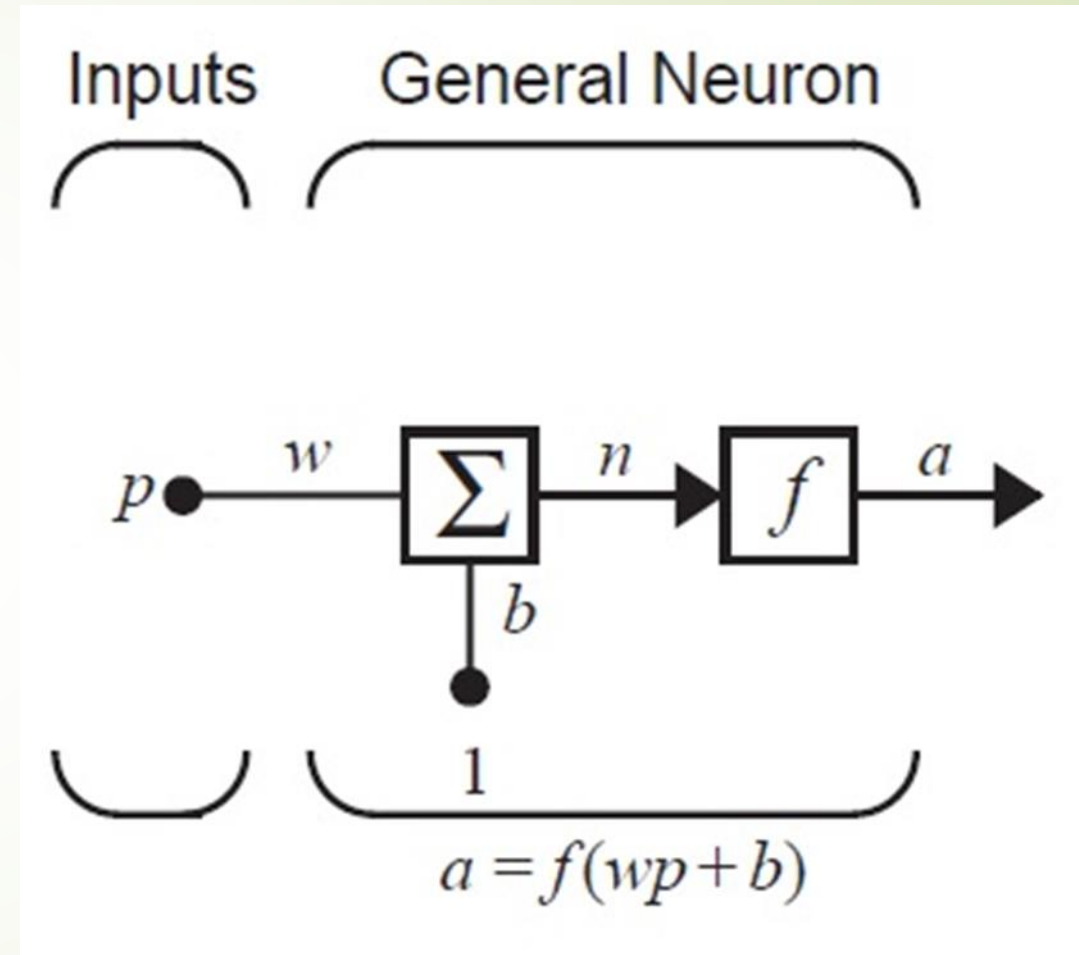




# Conceptos básicos

Donde:

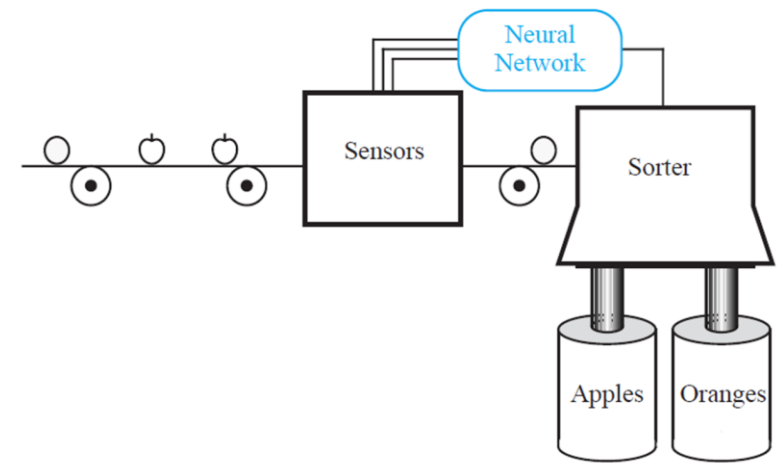
- $p$  = inputs
- $W$  = weights
- $b$  = bias
- $\Sigma$  = summation
- $n$  = result of summation
- $f$  = transfer function
- $a$  = output



# Funciones de transferencia

Name	Input/Output Relation	Icon	MATLAB Function
Hard Limit	$a = 0 \quad n < 0$ $a = 1 \quad n \geq 0$		hardlim
Symmetrical Hard Limit	$a = -1 \quad n < 0$ $a = +1 \quad n \geq 0$		hardlims
Linear	$a = n$		purelin
Saturating Linear	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n \leq 1$ $a = 1 \quad n > 1$		satlin
Symmetric Saturating Linear	$a = -1 \quad n < -1$ $a = n \quad -1 \leq n \leq 1$ $a = 1 \quad n > 1$		satlins
Log-Sigmoid	$a = \frac{1}{1 + e^{-n}}$		logsig
Hyperbolic Tangent Sigmoid	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$		tansig
Positive Linear	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n$		poslin
Competitive	$a = 1 \quad \text{neuron with max } n$ $a = 0 \quad \text{all other neurons}$		compet

# Ejemplo



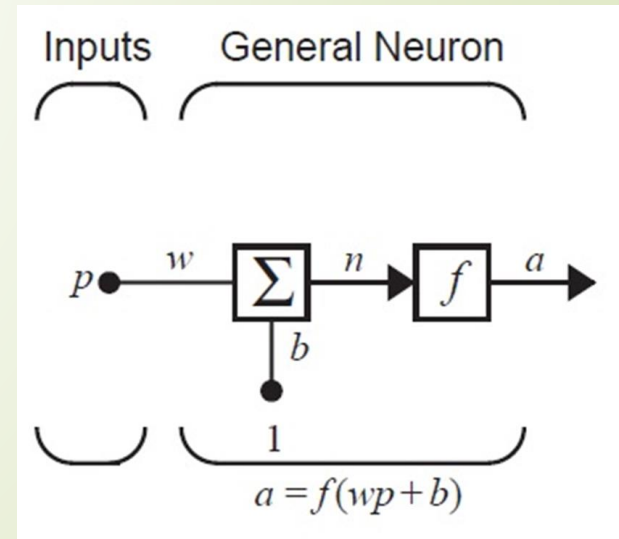
- Clasificar manzanas y naranjas de acuerdo a su forma, textura y peso.

- $p = \begin{bmatrix} 0.9 \\ 0.3 \\ 0.2 \end{bmatrix}$ ,  $w = [0 \ 1 \ 0]$ ,  $b = 0$

- *targets*:  $p_1 = \begin{bmatrix} 1 \\ -1 \\ -1 \end{bmatrix}$  (naranja)  $p_2 = \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}$  (manzana)

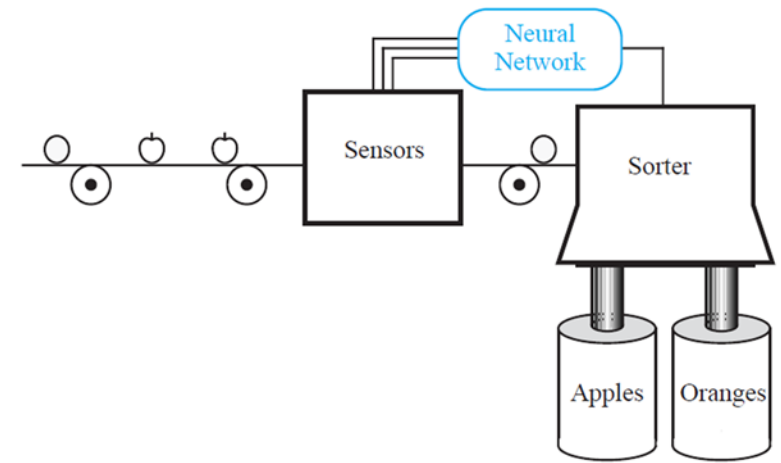
- $a = -1$  naranja  $a = 1$  manzana

- $f = \text{hardlims}$





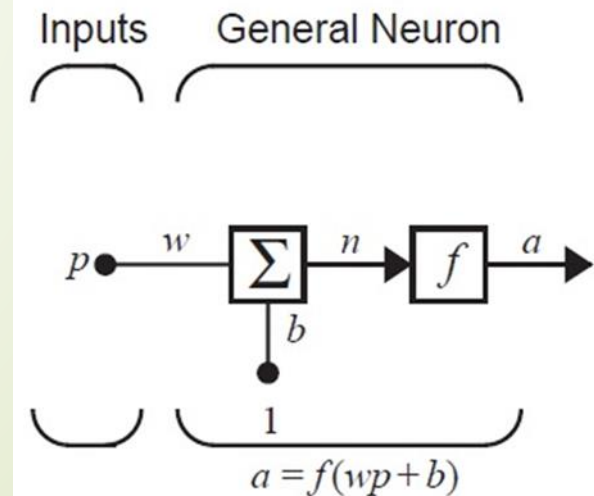
# Ejemplo



$$\rightarrow a = \text{hardlims} \left( [0 \ 1 \ 0] \begin{bmatrix} 1 \\ -1 \\ -1 \end{bmatrix} + 0 \right) = -1 \text{ (naranja)}$$

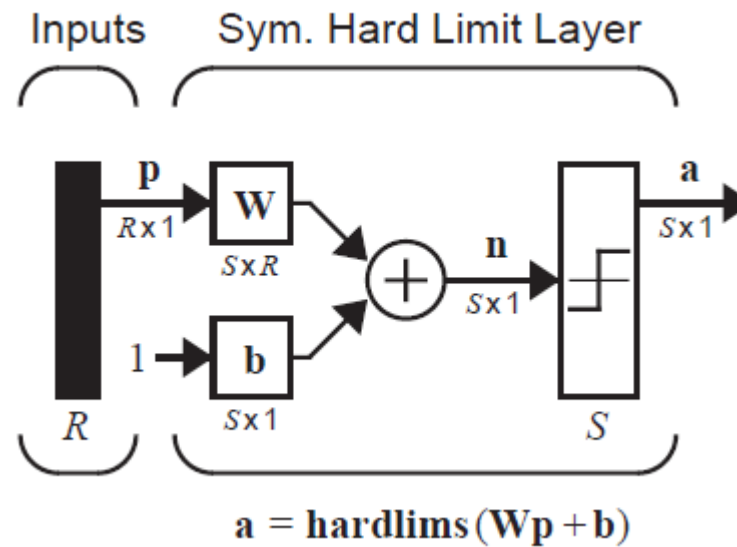
$$\rightarrow a = \text{hardlims} \left( [0 \ 1 \ 0] \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} + 0 \right) = 1 \text{ (manzana)}$$

$$\rightarrow a = \text{hardlims} \left( [0 \ 1 \ 0] \begin{bmatrix} 0.9 \\ 0.3 \\ 0.2 \end{bmatrix} + 0 \right) = 1 \text{ (naranja)}$$



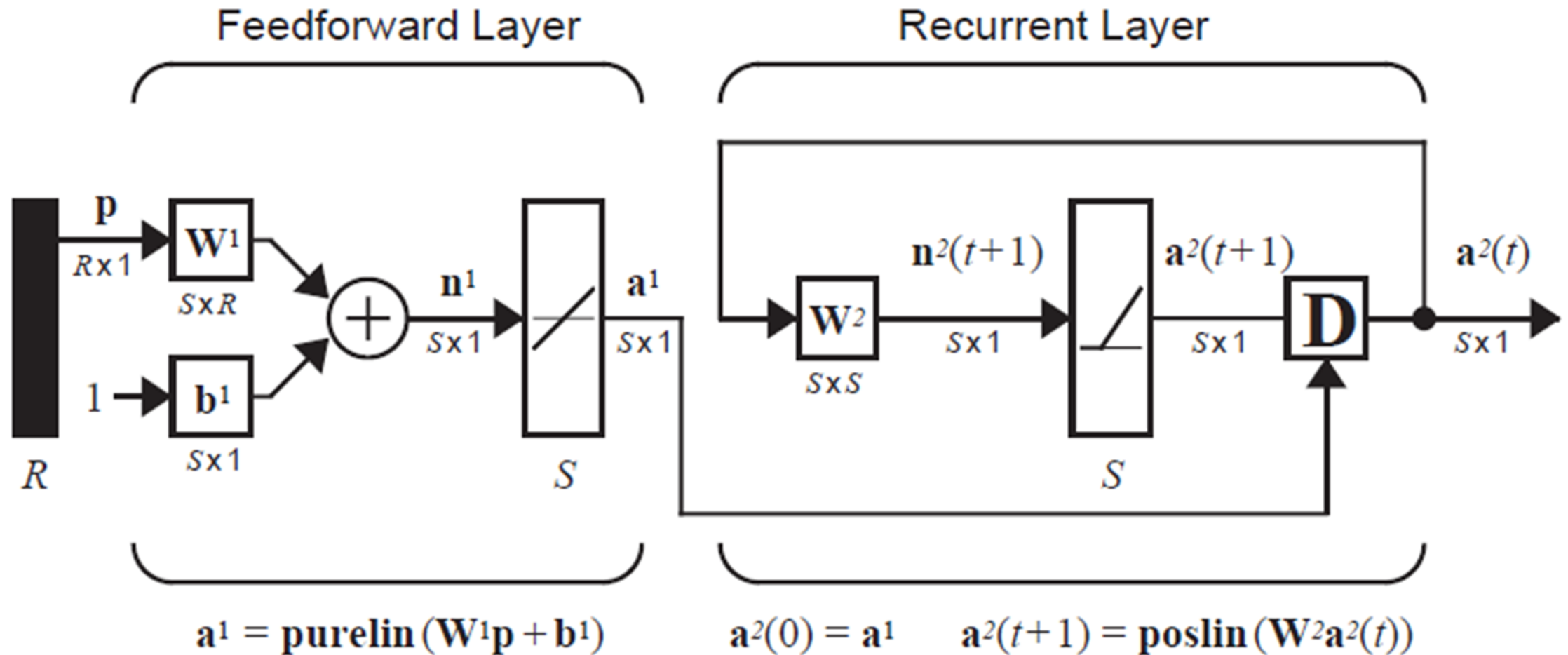
# Tipos de redes

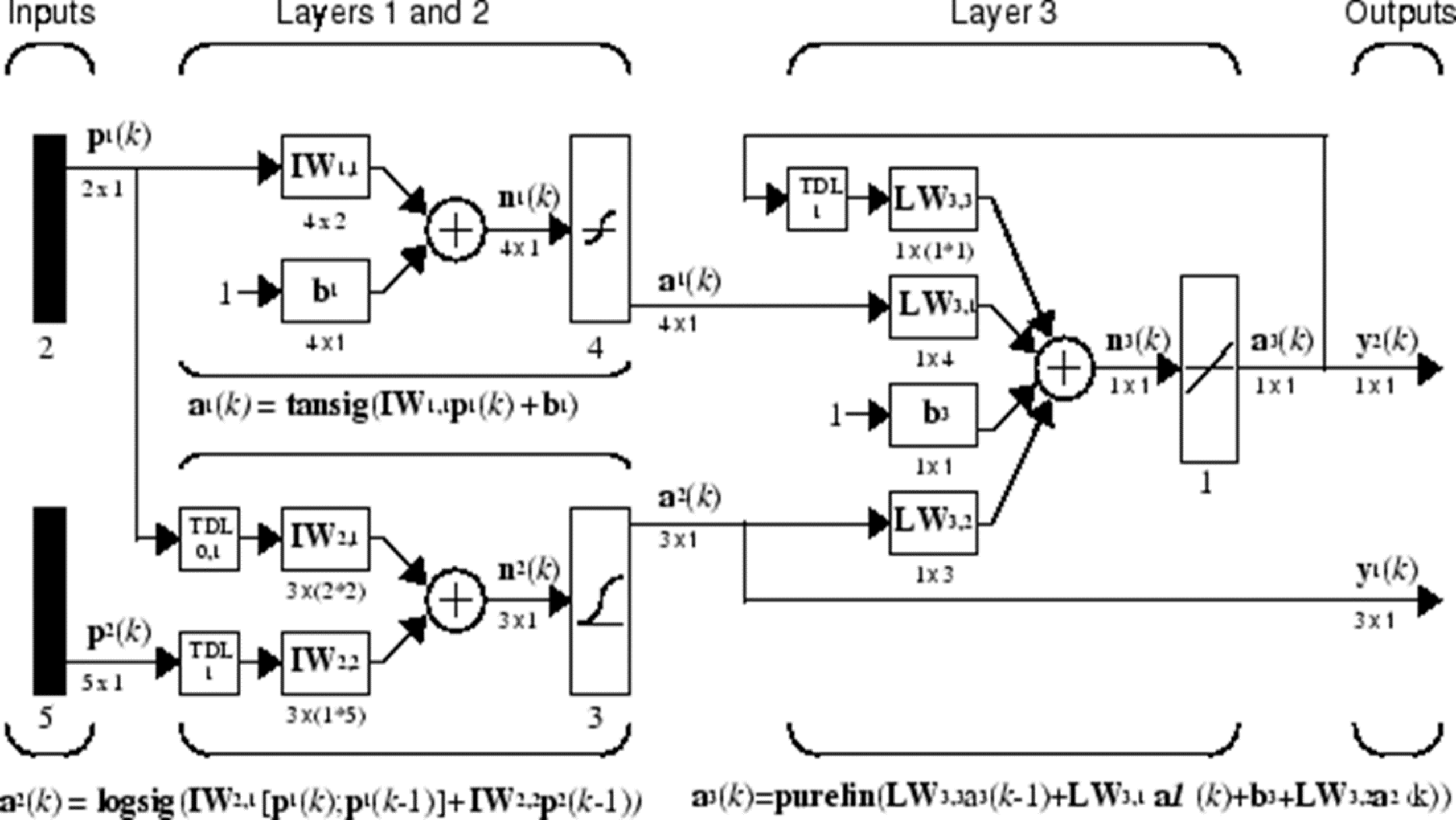
## Perceptrón



# Tipos de redes

## Hamming Network







# Uso en TAOSII







# Experimentación y resultados





# Parámetros considerados de los datos

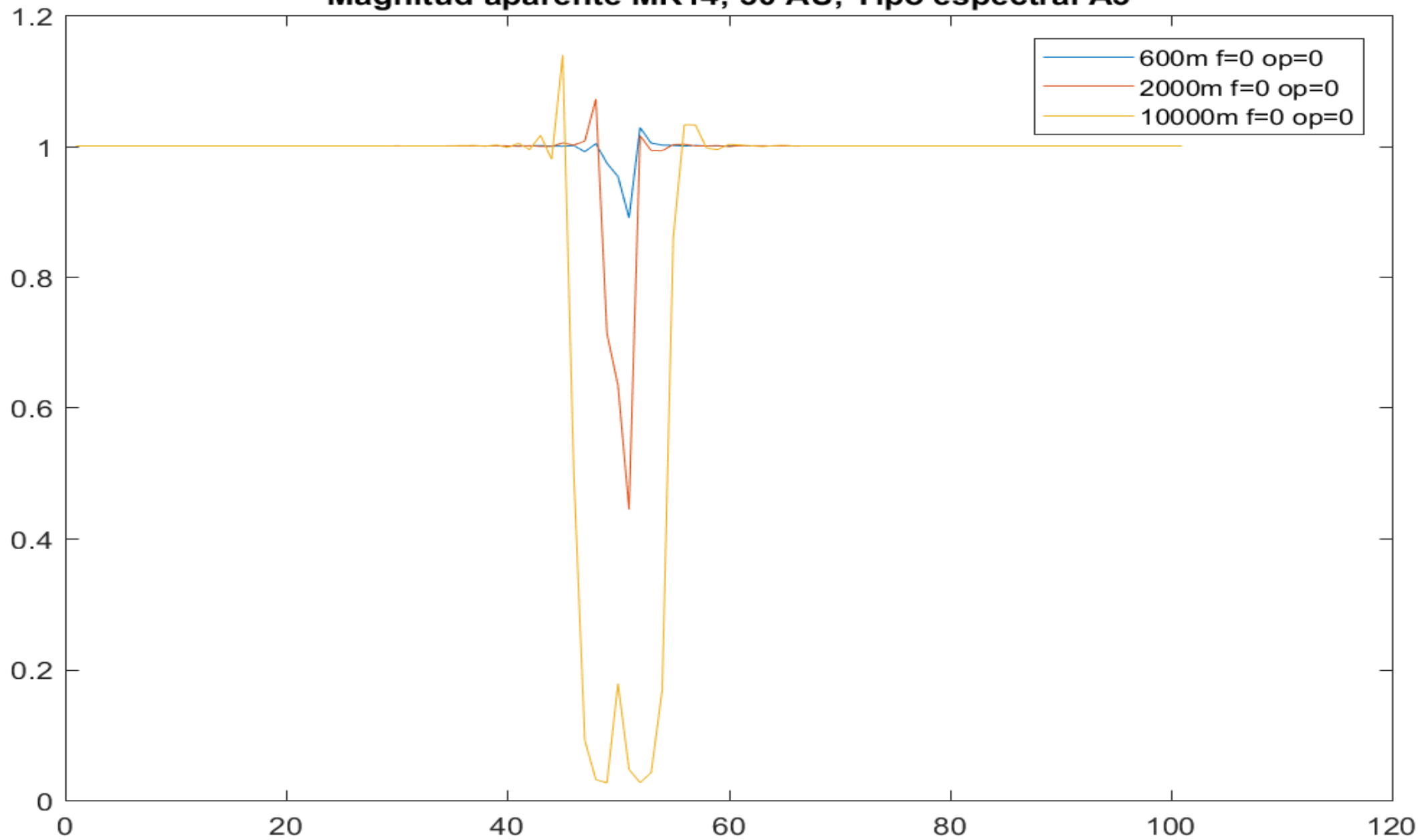
De la estrella de fondo:

- Magnitud aparente
- Tipo espectral

Del objeto a detectar

- Ángulo de oposición
- Diámetro
- Distancia
- Factor de impacto

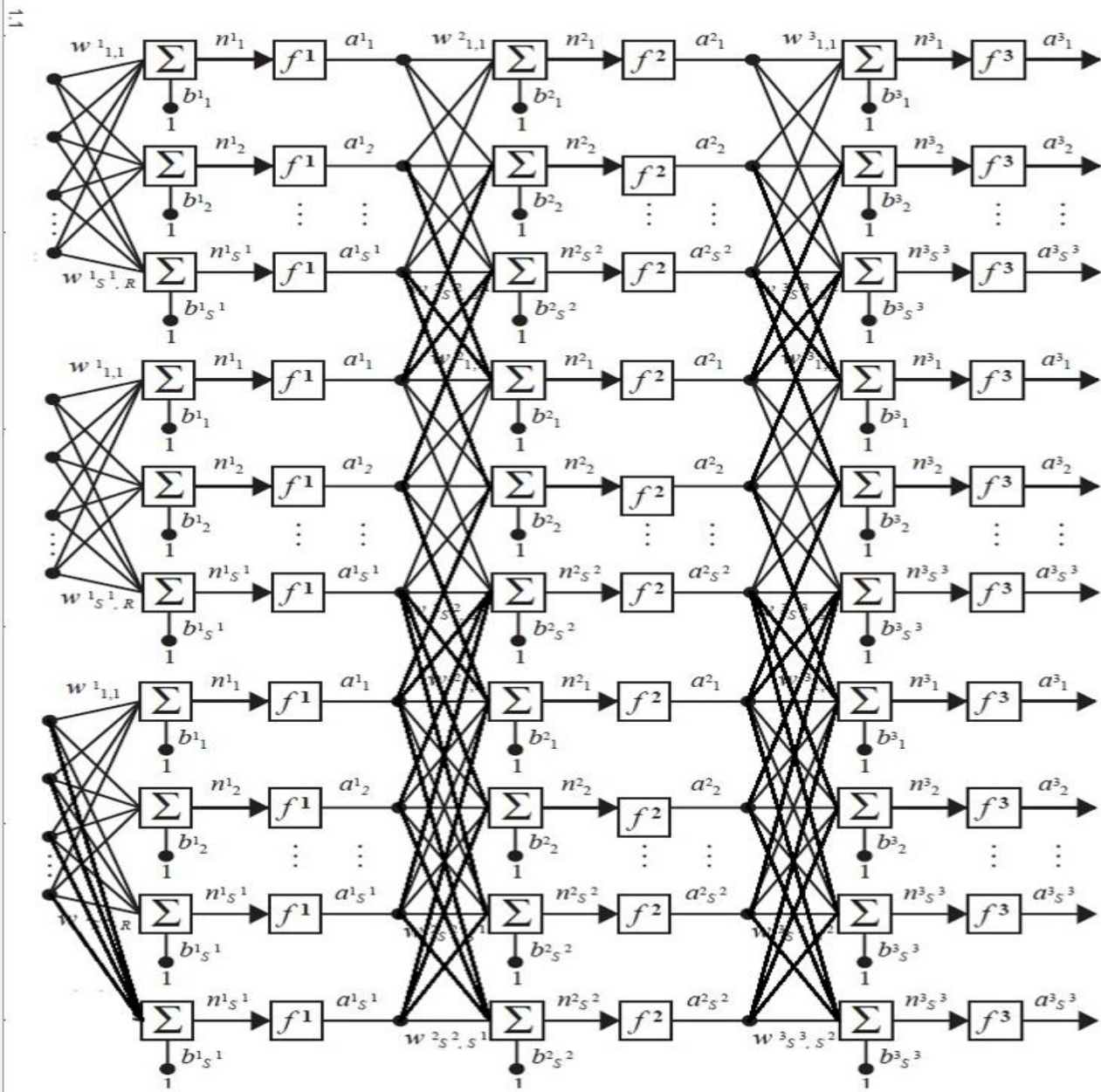
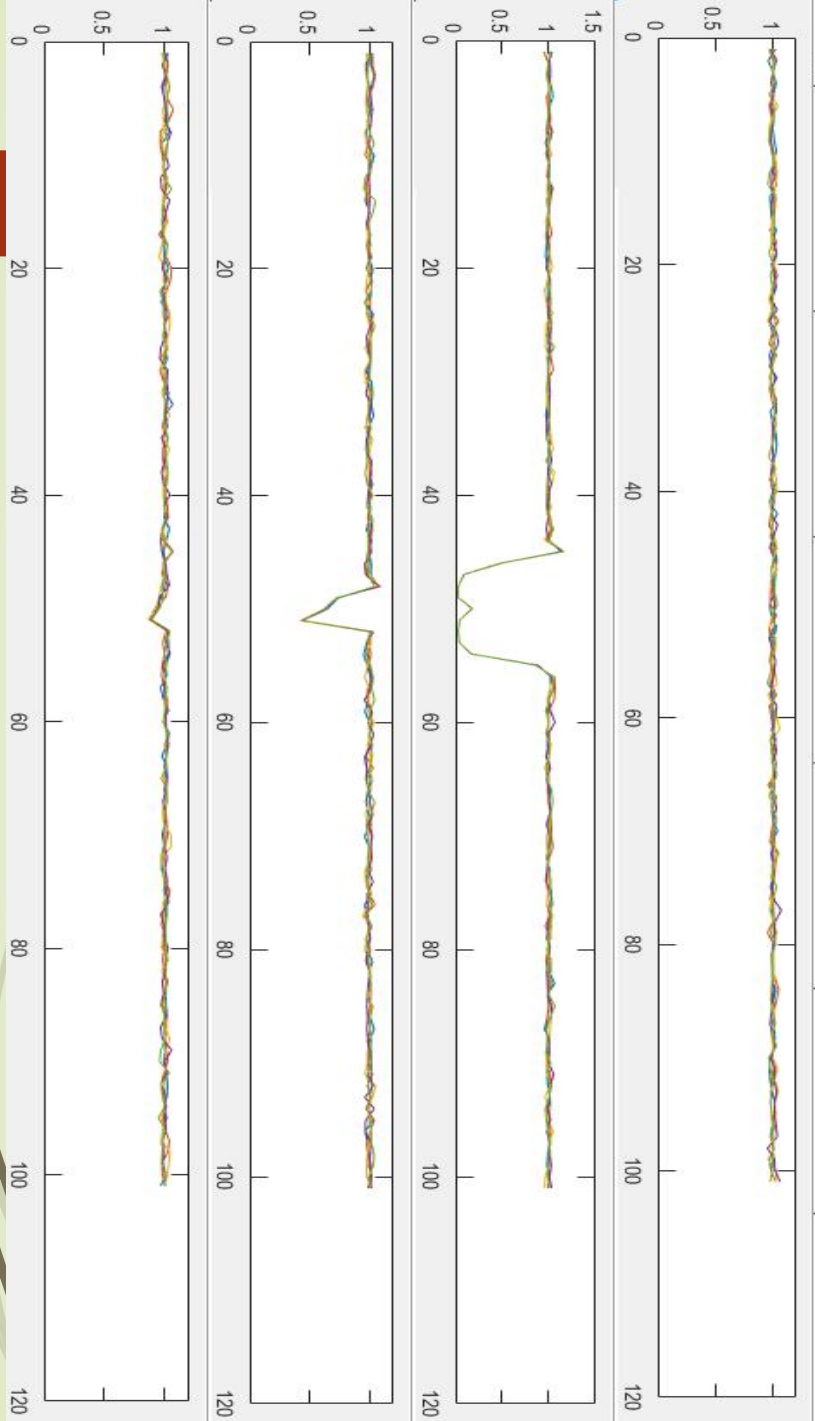
Magnitud aparente MR14; 30 AU; Tipo espectral A3





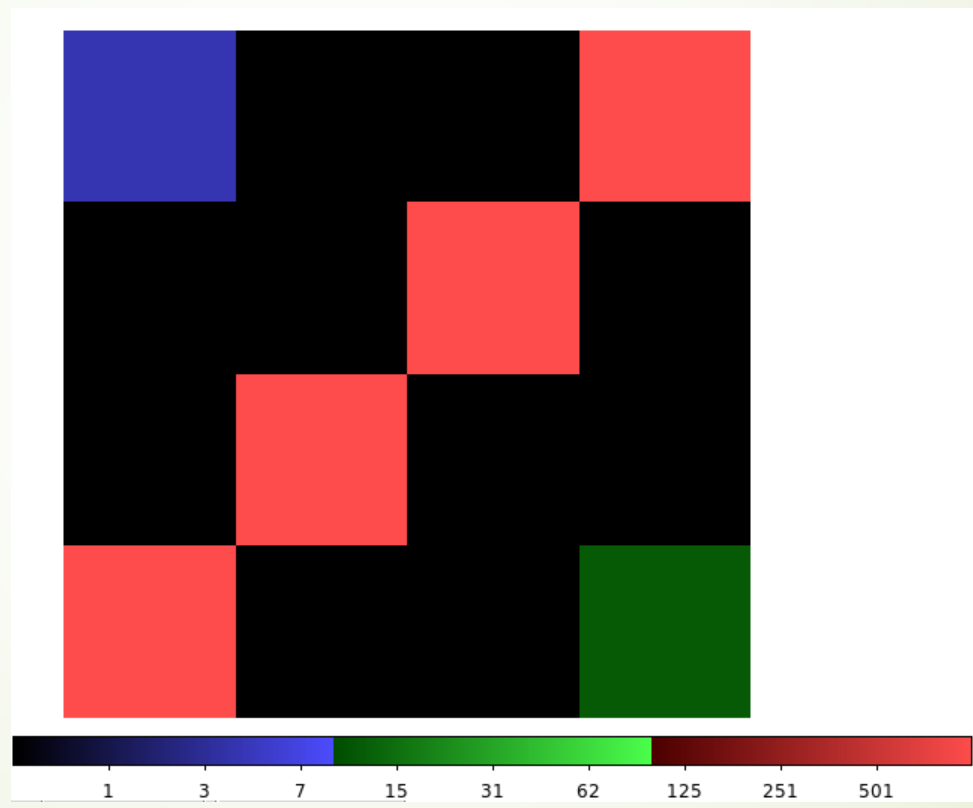
# Parámetros de las redes diseñadas

- Tipo: red de reconocimiento de patrones
- Tamaño entrada: 101x4000 (4 clases )
- Capas: 3
- Neuronas por capa: 10
- Función de transferencia: sigmoide logística (logsig)
- Método de entrenamiento: gradiente conjugado escalado



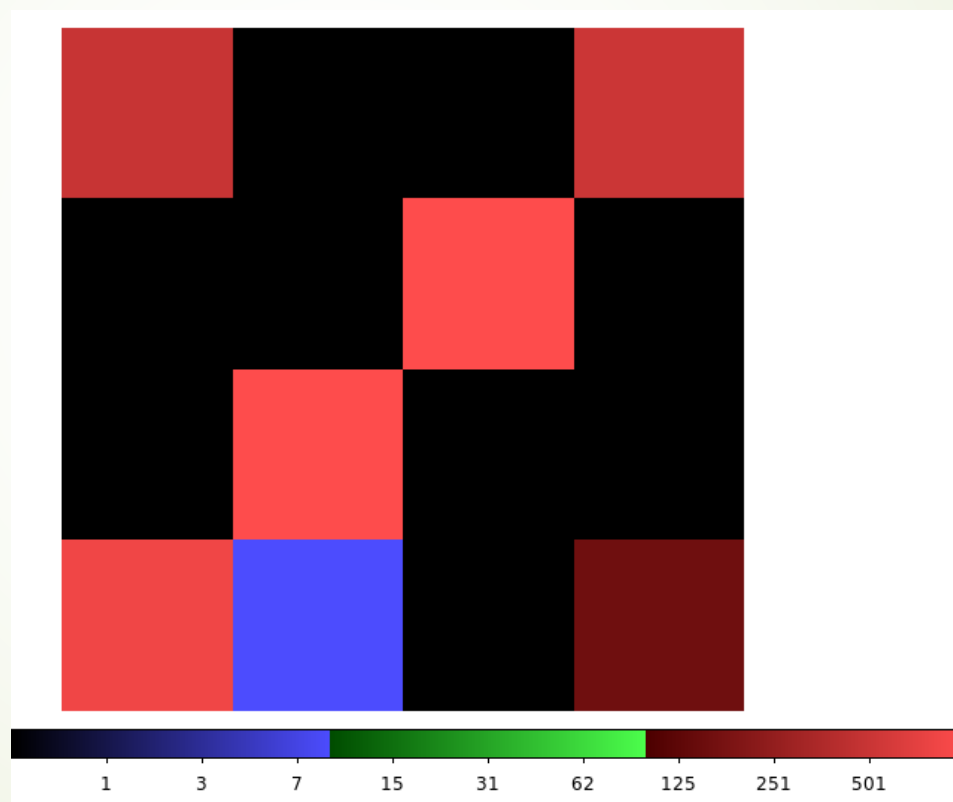


# MR14





# MR16

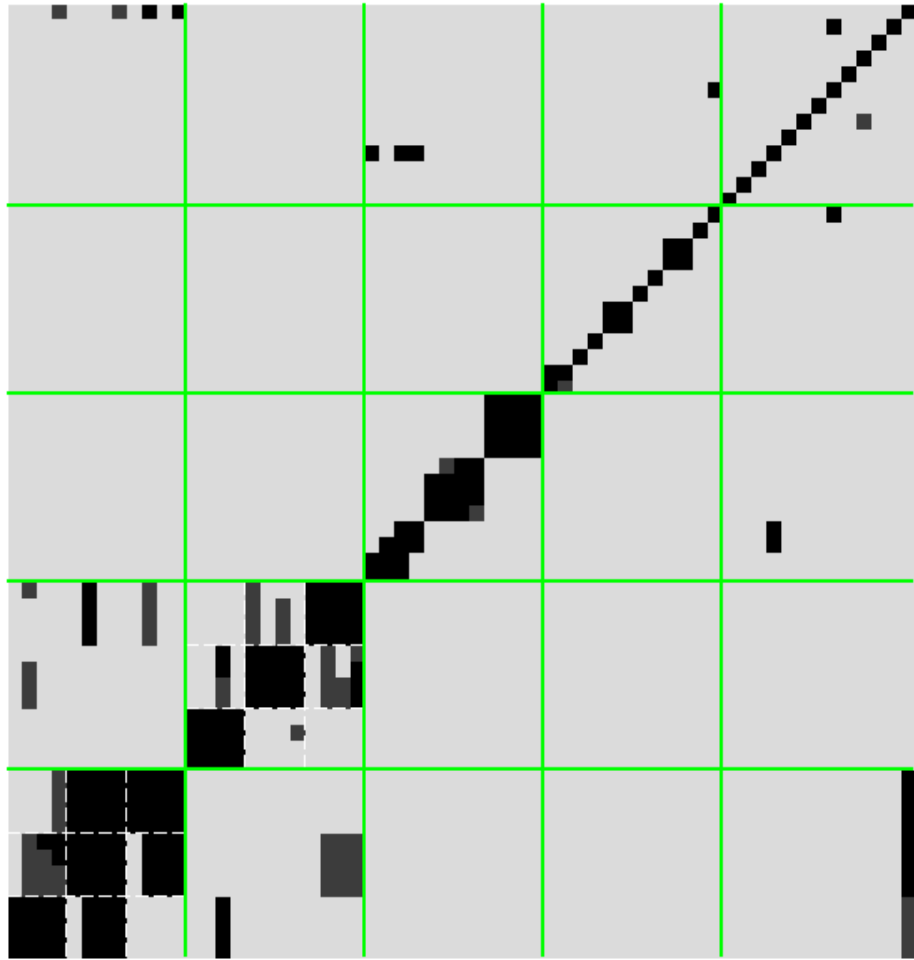




# Parámetros de las redes diseñadas

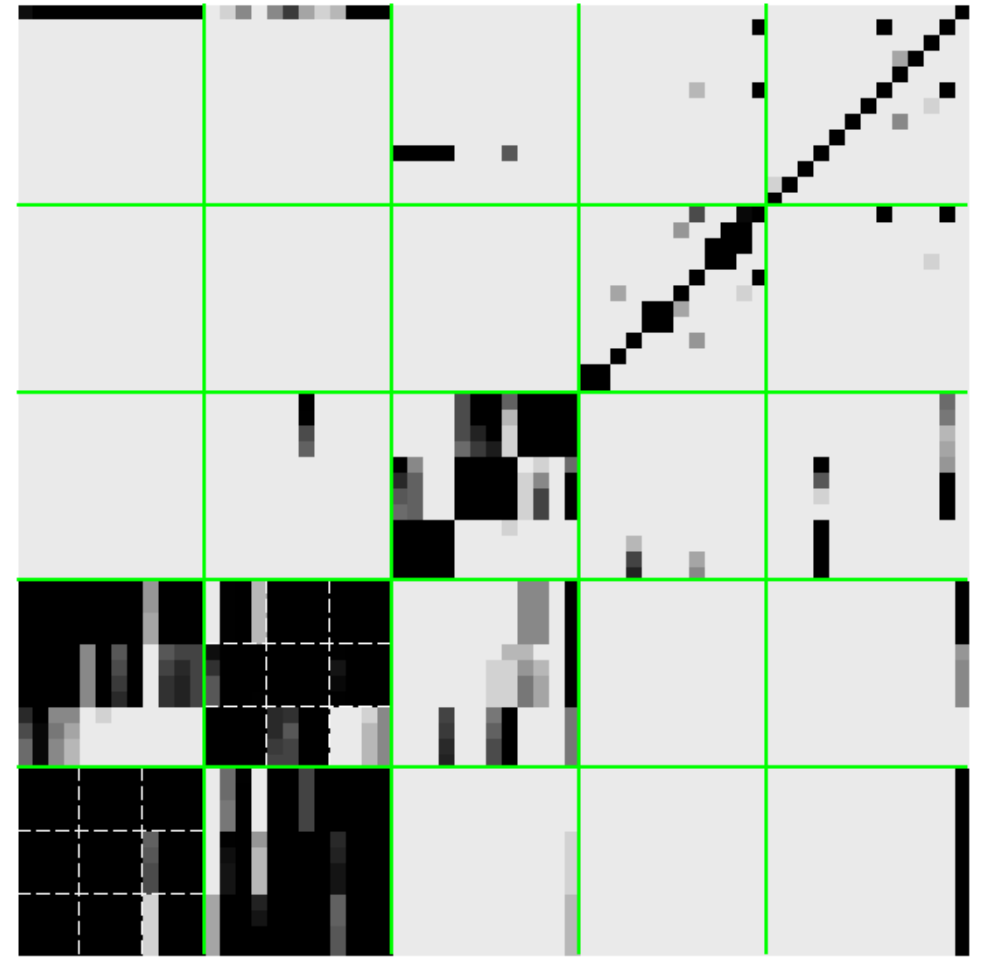
- Tipo: red de reconocimiento de patrones
- Tamaño entrada: 101x61000 (61 clases )
- Capas: 4
- Neuronas por capa: 100-200
- Función de transferencia: sigmoide logística (logsig)
- Método de entrenamiento: gradiente conjugado escalado

# Entrenamiento



# MR12

# Prueba

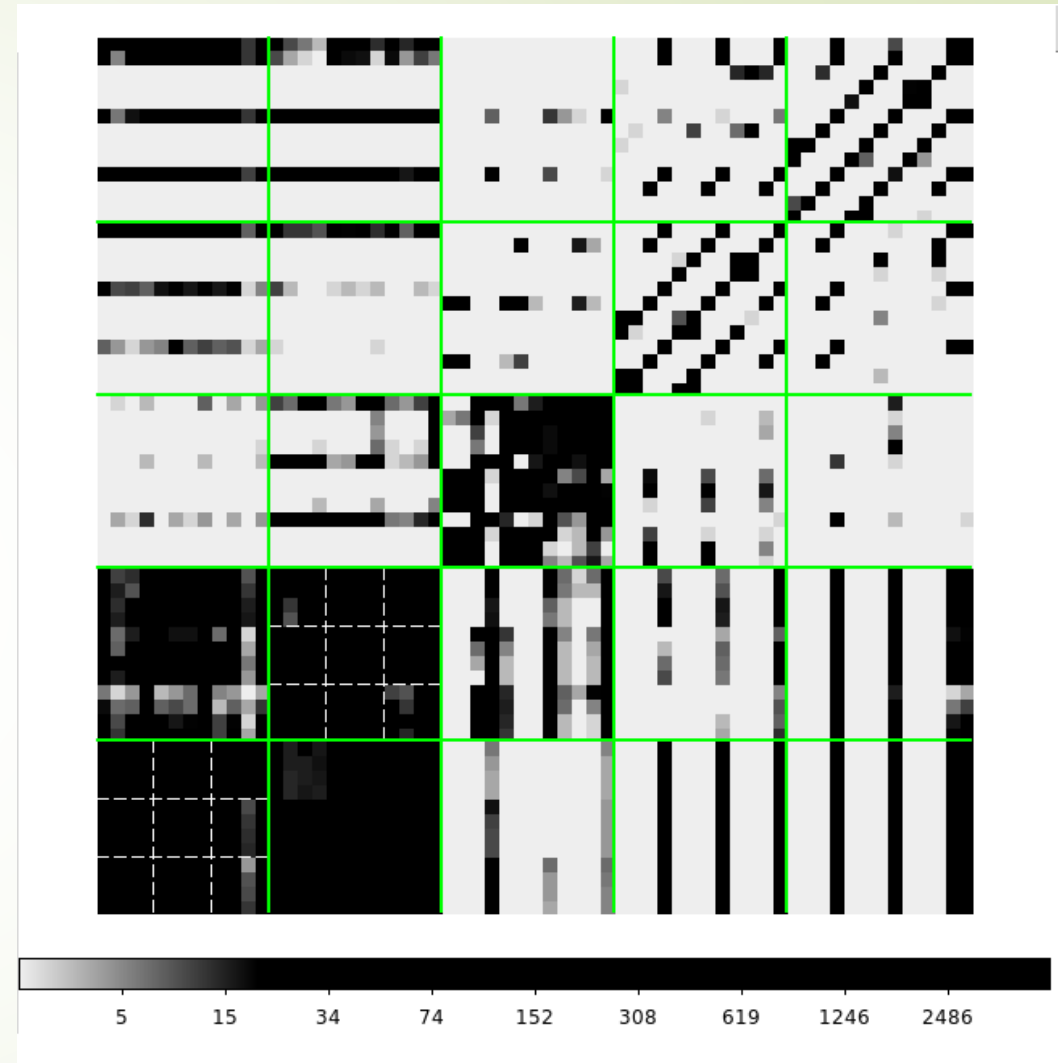
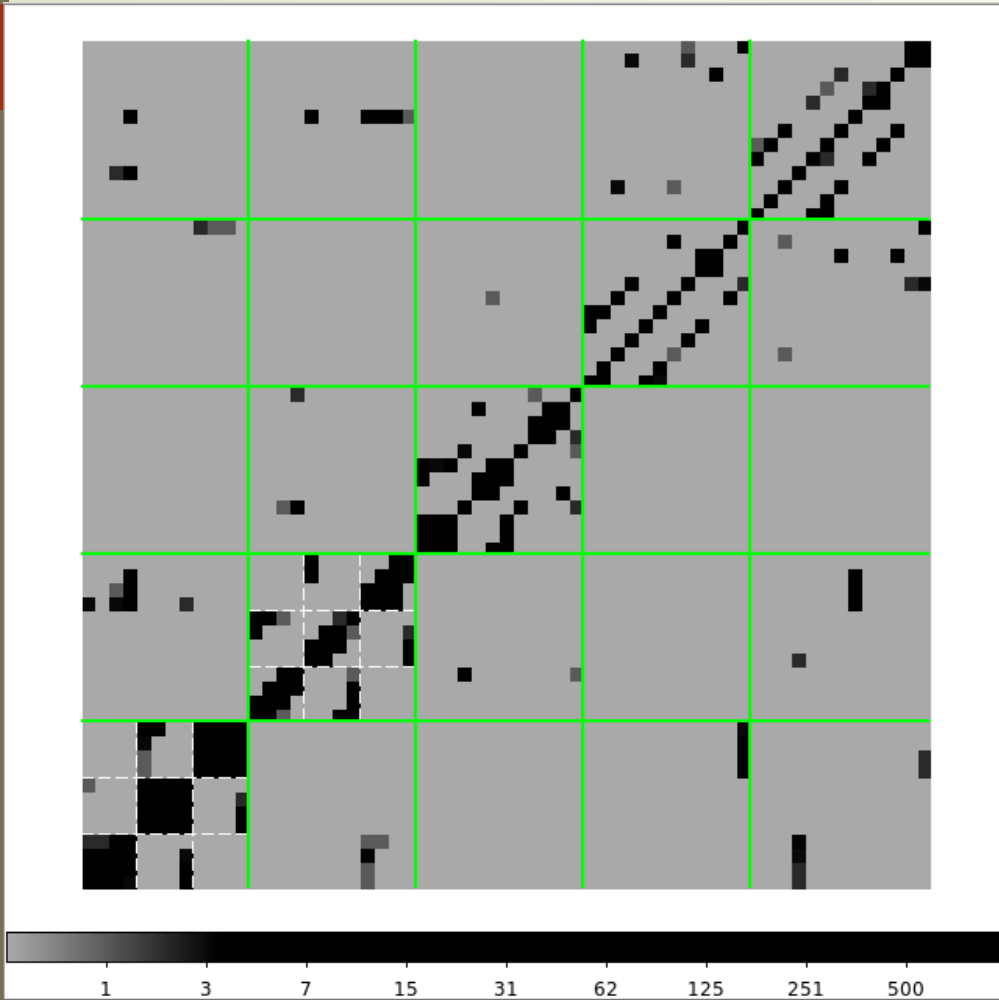


Factor de impacto	0, 0.25, 0.5, 0.75
Distancia (AU)	30, 40, 50
Diametro (km)	0.6, 1, 2, 5, 10

# Entrenamiento

MR16

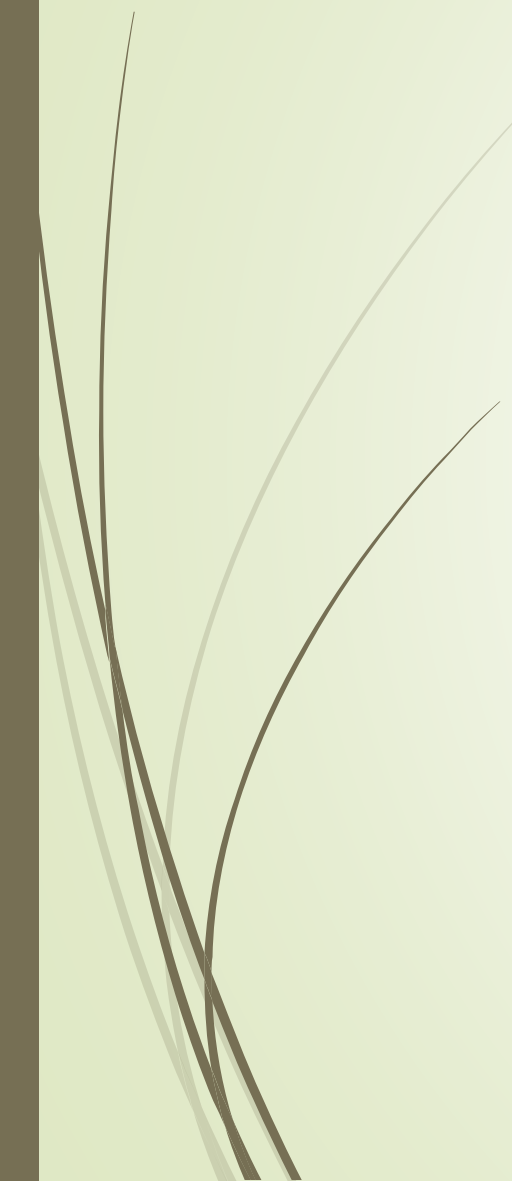
# Prueba



Factor de impacto	0, 0.25, 0.5, 0.75
Distancia (AU)	30, 40, 50
Diametro (km)	0.6, 1, 2, 5, 10



# Conclusiones

- Las redes neuronales funcionan mejor para objetos de diámetro más grande.
  - Aunque a pequeña escala se obtuvieron resultados casi perfectos, a gran escala se requiere de un extenso análisis sobre el comportamiento de la red con mayores dimensiones.
  - Las redes neuronales son una gran herramienta de inteligencia artificial que vale la pena seguir trabajando.
- 



# Referencias



- ▶ Ponce, Pedro. (2010). *Inteligencia artificial con aplicaciones a la ingeniería*. México: Alfaomega Grupo Editor.
- ▶ Hagan, Martin. Demuth, Howard. Beale, Hudson. De Jesus, Orlando. *Neural Networks Design*. [hagan.okstate.edu/nnd.html](http://hagan.okstate.edu/nnd.html)
- ▶ Hagan, Martin. Demuth, Howard. Beale, Hudson. *Neural Network Toolbox User's Guide*. [www.mathworks.com](http://www.mathworks.com)